ESD-TR-67-452

COST REPORTING FOR DEVELOPMENT
OF INFORMATION PROCESSING SYSTEMS

Edward A. Nelson
Thomas Fleishman

11 April 1967

COMPUTER DIRECTORATE
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

AD0657793

LEGAL NOTICE

OTHER NOTICES

ESD-TR-67-452

COST REPORTING FOR DEVELOPMENT
OF INFORMATION PROCESSING SYSTEMS

Edward A. Nelson
Thomas Fleishman

11 April 1967

COMPUTER DIRECTORATE
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
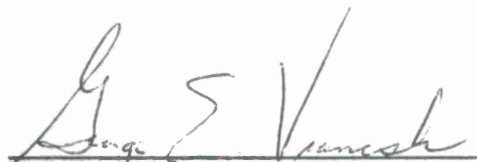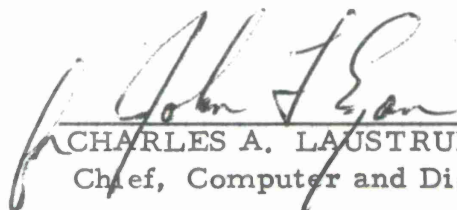L. G. Hanscom Field, Bedford, Massachusetts

FOREWORD

This report was prepared by the Programming Management Project at System Development Corporation (SDC), under contract with the Directorate of Computers, Electronic Systems Division, Air Force Systems Command. The research in the Programming Management Project is directed generally toward the development of guidelines, standards, and techniques that contribute to improved management of computer programming activities. Previous work has included an investigation of the various factors that influence the expenditure of resources on computer programming applications, the development of guides to estimate these expenditures, and guides for planning large and small applications. This report deals with the problem of collecting information on computer programming, during the progress of the job. Such information may be used for the managerial control of current computer programming work, and the building of a data base from which new management tools can be developed. As such, the work represented here is intended as an improvement upon an earlier effort reported in SDC TM-2934/000/02 (1).

Victor LaBolle, Leader of the Programming Management Project since its inception in 1962, contributed substantially to the integration and textual content of this document. The authors are also indebted to G. Weinwurm, L. Searle, N. Willmorth, C. Starkey of SDC for their review and comments on the preliminary draft, and particularly to K. Petersen for his valuable contributions to the final product.

This Technical Report has also been published by SDC as TM-3411/000/00, "A System for Collecting and Reporting Costs in Computer Program Development."

This Technical Report has been reviewed and is approved.

GEORGE E. VRANESH
Capt, USAF

CHARLES A. LAUSTRUP, Col, USAF
Chief, Computer and Display Division

ABSTRACT

This report describes a system for the collection and reporting on data on the resources expended in the production of computer programs. The system is intended to: (1) provide information to facilitate management control during the progress of a computer programming effort; (2) build a data bank from which better cost-estimating relationships and planning tools can be developed; (3) accomplish the above with a minimum of interference with operating personnel. The report was designed to provide sample materials necessary for the implementation of cost reporting in any organization in which computer programming is performed; it includes a description of the steps that constitute the computer programming process, the kinds of personnel who would be involved in the cost-collection and -reporting system, a recommended work flow and suggested forms for use in data collection and reporting, a work breakdown structure for associating costs with activities, and a brief discussion of the relationship of this system with several existing Department of Defense management procedures.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

SECTION I

INTRODUCTION

Like many emerging industries, computer programming has few standards and
guidelines to help managers, buyers, and analysts in planning, controlling,
and evaluating the work to develop computer programs. Aside from the youth-
fulness of the field, this dearth of benchmarks reflects the relatively
small effort that has been devoted to systematic analysis of computer
programming as a process. To help improve this situation, this report is
intended to provide a system for collecting valid, reasonably comparable
data that relates work content to the costs incurred.

In this section, the following topics are discussed:

   . Summary of the contents of this report

   . Statement of the problem

   . Scope of this particular report

   . Audience of this report

1. <u>Summary</u>. This report describes a system for the collection and reporting
of data on the resources expended in the production of computer programs.
The data collected will provide information to help managers control the
computer programming process, and build an experience reservoir from which
improved management tools can be developed. A total system approach is
taken, recognizing the existence of other possible interfacing elements
such as hardware development; however, only costs directly associated with
the computer programming process are at issue herein.

Three considerations are paramount in this suggested cost-collection and
-reporting system: a uniform breakdown of the total information processing
system development process into steps; collection of direct costs at the
source level, and association of these costs to the appropriate step within
a logical hierarchy based on activities performed; and assignment of the
major burden of cost collection to a staff, rather than a line, operation.
Collection of costs on all computer programming efforts within an established
work breakdown structure provides the uniformity that permits the comparison
of various developmental projects. A hierarchy, starting with direct costs
at the lowest point where such costs are incurred, allows the manager to
trace the causes of cost overruns or schedule slippages, and provides data
that can be used to answer the widest variety of questions about the process
of information system development; such data are not distorted by arbitrary
indirect cost allocations. The assignment of the major portion of the

cost-collection and -reporting function to a staff rather than to a line operation enables the collection process to proceed with minimum interference to line personnel.

In this report, the data-collection process is described in considerable detail. The description includes the enumeration and definition of data items to be collected, the personnel involved in the collection process, and the work flow or sequence of data collection. The data items are presented in a series of sample forms; with slight modifications, these forms, or forms similar to them, would be adequate for most information processing system development efforts.

The proposed collection and reporting system provides flexibility to the users in two ways: first, the level of detail available allows the use of optional portions of the data-reporting hierarchy; second, the collection forms are grouped in this report according to their intended purposes. That is, the primary source data-collection form, the suggested reports for management, and a set of forms containing data for research purposes, are discussed separately. This grouping makes it easy for an organization that wishes to collect data only for management control to ignore the majority of data items and forms. Also, the capability is provided for collecting costs associated with specific activities (e.g., report writing, conference attendance, supervision) if a manager desires to single out such activities for special attention or surveillance.

2. Statement of the Problem. The objectives of this cost-collection and -reporting system are: (1) to provide information to enhance management control during the progress of a computer programming effort; (2) to build a data bank from which better cost-estimating relationships, planning tools, and other management guidelines can be developed; (3) to accomplish the above with a minimum of interference with operating personnel.

Managers of computer programming have few proven and reliable ways of relating costs to product characteristics. The development of suitable management tools has been obstructed by a lack of common terminology, by cost-collection procedures that are oriented toward legal-accounting purposes, and by the fact that the technical constraints that impinge upon the computer programming process are not well enough structured and described to promote progress in this area. This report will not resolve all of these difficulties. But it does take a step toward such a solution; it presents a structure and set of procedures that can be used by most installations, and from which important advances in computer programming management can evolve.

3. Scope. The focus of this report is on the procedures, data items, and kinds of personnel involved in the collection and reporting of the costs of developing computer programs.

This is a generalized system, intended to be applicable in all kinds of computer programming efforts and organizations. It is recognized that many computer applications, particularly those developed for the Department of Defense, may also involve the development of hardware, facilities, man-machine interfaces, and other components whose design influences the computer programs; however, the system described in this report is specifically restricted to the identification and collection of only that portion of total system cost that has a direct information processing context. Other costs, applicable to the total system of which the information processing system is only a part, are beyond the scope of this work.

In addition to its major objective of collecting and reporting information on computer programming development, this report also provides a structure that includes a breakdown of the computer programming development process into nine steps, starting with preliminary planning, and culminating with information processing system maintenance. Also discussed are the system's compatibility with existing management systems in the government, such as the AFSCM 375 series, the procedures used by Air Force officers who manage the development of major electronic or weapons systems in Air Force System Project Offices and the PPBS, the Program Planning Budgeting System, used to identify and compare planned expenditures for major outputs of various government agencies as they relate national or agency goals and objectives.

In addition to the necessary tools to collect information, this report also presents a planning framework that can be used in conjunction with previous Programming Management Project work (2, 3) to develop the total structure required for control. For the worker investigating computer programming management generally, we present a number of typical questions that may be addressed by the data items suggested for collection.

Specifically excluded from this report are cost-estimating procedures to be used in planning (2, 3). Also excluded are procedures related to the correction aspects of management control. That is, the collection and reporting system provides information to alert the manager to impending problems in schedules and resources expenditures; however, the solutions to these problems--what the manager should actually do to correct an adverse situation--are not treated.

4. Audience of this Report. This report was intended for use by three types of personnel: the staff whose responsibility it is to collect and report data; the management; and personnel developing improved tools for the planning and control of computer programming.

The staff personnel most directly concerned with collecting, summarizing, and reporting data will need to examine this report in its entirety to understand fully the purposes and procedures of the proposed system. The suggested data-collection forms, or adaptations of them, will be the working tools of these personnel.

The reporting structure to assist in the control of ongoing computer pro-
gramming efforts was designed for two broadly defined levels of management.
The first level consists of those managers most concerned about the overall
progress of a particular, organized developmental effort. The responsibility
here would involve the successful completion of a single contracted-for
end item. The USAF System Project Office (SPO), or a Project Manager (as
an example of contractor personnel) would be typical of this level of
management. The second level of management addressed consists of those
managers whose purview includes more than one, and possibly unrelated,
developmental efforts. This responsibility involves an allocation of
resources, and perhaps the establishment of priorities, for all of the
work done within the organization in question. For computer programming,
various departments of Headquarters USAF (4), or a Director of Data
Processing (as an example of contractor personnel) would be typical of
this level of management. The reporting material intended for both of
these management levels is separately described in this report.

For those engaged in the development of better management tools and guide-
lines, this report suggests a number of questions that data collected by
this system could help to answer. In addition, the structure and content
of the data bank that would result from the implementation of the proposed
system is described.

This cost-collection and -reporting system was intended to meet the general
needs of the audience mentioned above. Specific tailoring of the system to
meet the special requirements of different organizations may be indicated
in some instances, particularly in regard to research efforts and reporting
to management. Also, the data items cited may have to be defined in terms
of the vocabulary and forms already in use by the organizations that install
the proposed system. However, substantial modifications of the basic cost-
collection structure would inhibit the comparison of different programming
efforts, and should not be necessary.

4

SECTION II

BACKGROUND

This section provides a background to assist the reader in interpreting the proposed cost-collection and -reporting system. Briefly discussed are the following:

- The needs of managers who are responsible for computer program development

- Work done earlier to satisfy these needs

- The similarities and differences with past work at System Development Corporation

- Considerations for use of the system and benefits that can be derived by adopting it

1. Needs for Management Tools. Managers who are responsible for development of information processing systems need improved tools to help plan and control these efforts. Specifically, managers involved in computer program development would like the means to translate requirements for new or changed information processing systems and their computer program components into development plans with reasonably accurate estimates of schedules and costs for these efforts. As a foundation for control, these managers also need ways to compare the progress of the actual work on the products with detailed milestones in these schedules and the actual expenditures with those projected over time. Further, to make such plans, managers would like to be able to assess more readily the impact of differences in development resources--personnel, programming tools, computers and their configurations--as well as environmental factors, on their costs and schedules.

Several factors have slowed the evolution of tools and guidelines to satisfy these management needs:

- The basic product in computer programming is intangible and difficult to relate to costs.

- The technology in automatic data processing changes at a rapid rate, so experience is difficult to generalize and transfer.

- Few numerical data have been collected to quantify experience.

- Many of the numerical data that have been collected are not uniform so that quantitative comparisons and analyses could not be made with high confidence.

- Also, these numerical data have usually been gathered sometime
  after the work has been completed with some probable loss in
  data reliability.

- No standards exist to provide a framework for numerical analyses,
  standards that have been widely accepted and applied to describe
  requirements, products, resources, and the process whereby
  computer programs and information processing systems are developed.

2. Work to Meet These Needs. Recently, the decision-making problems that face
managers in computer programming and the buyers of the resulting products have
been identified more clearly. A landmark in this area, Brandon's Management
Standards for Data Processing (5), describes techniques for establishing
standards for methods and, subsequently, performance standards for the men and
machines used in computer program development. Such standards are aimed at
improved management control, cost estimation, and cost control, particularly
in the field of business data processing. Also, the Federal Government has
been addressing questions on how to plan, control, and evaluate computer
programming efforts. For example:

- The United States of America Standards Institute (USASI--formerly
  ASA) has been working to develop standards for tools such as
  programming languages.

- The Bureau of the Budget, the General Services Administration,
  and the National Bureau of Standards, starting with the
  formulation of policies on computer acquisition and use, have
  now begun to pursue standards for computer programming (6).

- NASA and the Electronic Systems Division of the Air Force
  Systems Command have sponsored projects at Bellcomm and System
  Development Corporation to help develop descriptions for the
  process of computer programming in terms of milestones that
  can be integrated into broader guidelines, e.g., AFSC 375
  Manuals, that existed earlier for the management of space and
  electronic system development.

- The Electronic Personnel Research Group at the University of
  Southern California, under an Office of Naval Research contract,
  has been developing task profiles and proficiency tests for
  various levels of programmers.

In company with such efforts, the Programming Management Project (PMP) at
System Development Corporation has been working since 1962 identifying and
developing aids that would permit managers to save time and money and achieve
better products in computer programming. Originally sponsored by the Advanced
Research Projects Agency, the Project has since 1964 been under contract with
the Air Force Electronic Systems Division and has also been sponsored by the

Office of Naval Research. Project members have engaged in several kinds of activities:

- Surveyed experience (7)

- Described or modeled the process of computer programming (8)

- Developed guidelines for detailed planning of computer programming efforts (2)

- Analyzed numerical data to derive equations for cost estimation (3, 9, 10, 11, 12, 13)

This cost-collection system set forth in this document has evolved from the cost analysis work.

To help managers make better estimates of costs for computer programming, Project members began exploratory work in 1964 to derive estimating equations using actual experience data as inputs to the analysis (9). Like the work to develop the cost-collection system, this earlier work was done under contract with the Air Force Electronic Systems Division, Directorate of Computers. Project members pioneered in this effort to gather and analyze numerical data on costs and probable cost factors for computer programming. The equations and planning factors derived in these analyses are rules for using numerical values for cost factors that characterize the requirements, resources, and environment for a computer programming effort to calculate estimates for costs such as manpower, measured in man-months, and computer time, measured in hours. These estimating relationships are intended to help the manager plan a computer program production effort in the early stages of computer programming, for example, before computer program design begins. The results may also be used by managers to evaluate completed efforts by comparing actual costs with estimates in a framework provided by the derived equations.

The results obtained in these analyses have large standard errors: managers cannot expect very accurate forecasts when using them. Although this lack of precision may accurately reflect the state of the art in the wide variation in costs and cost factors that occur in practice, project members felt that part of the spread could be attributed to variation in the reliability of data values. This uncertain reliability stems from (1) differences in the interpretation of the questions asked to obtain the data and (2) the collection of the data after the projects were completed with the attendant reliance on imperfect recall.

The use of a cost-collection system such as the one proposed here is intended to increase the reliability of the data to be used in the analysis of costs by collecting the data while the work is under way. Further, use of the same collection system for a variety of efforts in a single organization (or any group of them) can help assure comparable data by uniform interpretation of the

specific items required. As a result, these data are more likely to have the same meaning, and can be used to compare efforts while they are under way and as inputs to analyses to derive improved planning factors.

3. Similarities and Differences With Earlier Work. The cost-collection system described in this document is based upon the same premises as the earlier cost analysis work:

- Sufficient commonality exists in computer program development work to generalize and apply the same basic planning and control structure to a wide range of efforts and organizations.

- The basic problem in planning and control is to relate the technical content and development environment for a development effort to costs and schedules.

This cost-collection system is more comprehensive in its approach to costs and cost factors than the earlier work in PMP. The system is designed to span the entire range of activities in a life cycle of development, from conception to maintenance. Also, the system provides for the collection of dollar cost data whereas the earlier work was confined to analysis of basic resources--manpower (measured in man-months), computer time (measured in hours), and elapsed time (measured in months).

The design of the system recognizes that the same kinds of data can be used both in management control and in the development of improved planning factors, guidelines and/or standards.

The design philosophy is intended to ensure reliable cost data by collecting them in a timely manner and by requiring that they be verified. Recording of costs is common in both industry and government, but this system may differ from present practice in some cases, in that the costs are associated with products and the tasks and steps to develop these products rather than with organizations and gross categories such as labor and travel.

4. System Benefits. The system described here is intended to apply to a spectrum of different types of effort in computer program development in both government and industry. In this sense it is generalized and, if adopted by a particular organization, would require tailoring to fit its particular needs. Clearly, the widespread use throughout industry and government of a system could supply uniform data for purposes of comparison and analysis, and would benefit the entire ADP community by answering fundamental questions on costs as they relate to differences in techniques, type of application. However, widespread use of a uniform system would not be practical now and may not be feasible in the future. Even more mature industries with tangible products have not succeded in standardizing the collection of similar data. Therefore, for the near future, the question of benefits should be restricted to a particular organization.

8

The benefits that could be derived by using the system in that organization depend upon the present state of its control and planning methods. For example, an organization that presently prepares very detailed plans for computer program development efforts and uses such plans as a basis for control would probably find little value in the management control portion of this system. The only possible benefit that such an organization could receive from the proposed system would be improved understanding of the relationship between development cost and the products being developed and the steps and tasks performed in their development.

Among the types of organizations that do not plan and control at the level of detail recommended in this system, the one that would benefit the most would be a large organization involved in a spectrum of programming projects with different applications, a range of resources, and variety of machines. In this case, the adoption of a system such as this would provide a uniform planning structure and thereby immediately supply a way of comparing proposed and actual expenditures on various steps in computer program development from development to development and between types of application and different types of machine. In addition, by collecting the research data, a large organization can profitably conduct the analysis that would provide improved planning factors. For example, estimating relationships for cost and schedules could be derived for each step in the development process. Further, with sufficient amount of data, divisions can be made among these data to develop tentative classifications for computer program development efforts that depend upon their costs. For example, one could develop different equations for estimating costs corresponding to different applications or different machine types or different programming techniques (e.g., time-sharing, higher-order languages).

The majority of the data items identified in Section IV, Data Collected and Reported, are intended for use in research analyses to answer questions that bear on planning and evaluating computer programming efforts. The decision to collect all of these data should be made only if there are both an intent and a plan to expend resources for the analysis of these data and the use of the analytical results. Although almost all of these data are readily available at some place or point in computer program development, there is clearly a cost involved in collecting them and centralizing their location. Without a commitment to analyze the data and supply feedback in the form of planning factors to the operational managers, funds will be wasted in the collection process and personnel who contribute data will develop a negative attitude toward any further efforts to collect data.

Therefore, any organization considering the adoption and application of the cost-collection system should review carefully not only the data items but the research questions and compare them with the organization's needs for improved planning factors. As a result of such a review, new or revised research questions may be formulated and data items added or subtracted. In any such review, future needs for planning, control, and evaluation should be considered

9

Further, the managers examining the system should recognize that computer programming efforts may extend over long periods of time and that accumulation of accurate data for a sample large enough to permit an analysis with useful results may take several years.

SECTION III

A STRUCTURED APPROACH TO INFORMATION PROCESSING SYSTEM DEVELOPMENT

This section defines the steps in the development of an Information Processing System that can be used as a basis for planning, and for controlling the development of such a system. To supply more detail for control of larger efforts these steps are further divided into subordinate tasks. This breakdown of tasks and steps coupled with a way to group computer programming products forms a hierarchy. A coding scheme for identifying work within this hierachy is proposed; this provides a means by which the actual expenditure can be related to the kind of work being done and to a product. Finally, some relationships between the Cost-Collection and -Reporting System to other management information systems now used in DOD are identified.

1. The Steps in the Development of an Information Processing System. Although factors such as project size and organizational structure influence the extensiveness of planning and the preciseness of control desired in any given undertaking, some formal structure for planning and control is necessary to evaluate the experience on different computer programming efforts, and to evolve improved tools for the management of the computer programming process.

A basic requirement of a cost-collection system is that it be capable of associating all of the resources expended with the products produced, in this case a computer program component and associated user documentation. A computer program component is defined here as the complete sequence of machine instructions and routines necessary to solve a problem or perform a specified data processing function. A computer program component may be one of a set of other computer program components, all of which contribute to a larger data processing objective; this larger objective is served by what is herein called an information processing system.

The process of creating an information processing system involves the performance of a series of activities called steps. These steps constitute a breakdown of the system development process that is universally applicable to all types of data processing applications; however, in a given specific instance, the resources expended in certain steps may be negligible. The steps are analogous to a basic algorithm for formulating and solving a problem and testing a solution; either these steps, or similar ones, are commonly used in describing and planning computer work in both government and industry. Therefore, this step-planning structure represents a "natural" breakdown for this kind of work.

The minimum, formal reporting structure recommended in this collection procedure requires that resources expended (direct man hours, computer hours) be associated with the step appropriate to the activity performed.[1] The nine steps established for the Information Processing System development process are:

1. Information Processing Feasibility Analysis

2. Information Processing Analysis

3. Information Processing Design

4. Computer Program Design

5. Computer Program Coding and Checkout

6. Computer Program Functional Test

7. Information Processing Integration Test

8. Information Processing Installation and Implementation

9. Information Processing Program Maintenance

The names of these process steps use the prefix "computer program" to label a step that involves work identified directly with a specific computer program end product; the prefix "Information Processing" identifies steps that deal with other related products in the information system, such as total system cost-benefit studies (Step 1) or total system requirements specifications (Step 2). In this report the words Information Processing System may be interpreted as a range of entities--from a large system involving men, machines, computer programs to a single computer program. For those information processing systems with many components or subsystems, it is assumed that steps that are similar to the ones identified here for computer programming are pursued for each component.

In the following, each of the above nine steps is defined, and a list of possible tasks within each of the nine steps is suggested. This is not a complete list, but does provide an aid for more detailed planning and, consequently, sharper control of the development process. For control purposes, each task may be said to culminate in a milestone, or concluding event, whose completion becomes an objective measure of progress achieved to that date. Although only the collection of costs by steps is suggested as the minimum

---

[1] In this system, direct costs are associated with the steps for which they are incurred; indirect costs are summarized for the total organized effort and reports, but are not allocated to the individual steps.

mandatory breakdown, the collection of costs by tasks within steps obviously provides for more complete information; how this may be done will be discussed in the section on work breakdown structure. To help the reader who is familiar with System Project Office Procedures, each of the nine steps is related to the work sequence as identified in the AFSCM 375 series context. These relationships are not detailed; complete descriptions of the work to apply AFSCM 375 procedures to computer programming, are found in the documents in reference 14.

a. Information Processing Feasibility Analysis--Step 1. Definition: This step consists of the feasibility study of the proposed program. Based on a statement of the user's requirements, an estimate is made of the manpower, computer time, elapsed time, and other resources required for the project. Using these estimates, a summary project plan and cost-versus-benefits comparison are prepared. No more analysis of the proposed information system is done during this activity than is absolutely necessary for cost estimation and preliminary planning purposes.

AFSCM 375 Context: Feasibility analysis consists of the recognition of data automation requirements, conception of possible systems to meet these requirements, and the selection of the preferred system to be implemented. It includes the issuance of a Specific Operational Requirement, Advanced Development Objective, or Operational Support Requirement.

Tasks: Determine information system requirements and characteristics

   Determine organization of activities, including number and identity of any subsidiary efforts

   Select appropriate planning factors

   Estimate computer programming costs for each step or task

   Estimate costs other than computer programming

   Check reasonableness of estimates

   Prepare summary budget plan

   Determine costs of existing system

   Prepare cost-benefits evaluation of the proposed application

b. Information Processing Analysis--Step 2. Definition: This step consists of the detailed study of any existing system, and the formulation of the operational requirements for information processing in the proposed system.

AFSCM 375 Context: Information Processing Analysis begins in the Conceptual Transition Phase, after issuance of a Specific Operational Requirement (SOR), and Advanced Development Objective (ADO), or an Operational Support Requirement (OSR), and ends during Phase A, Prepare for Contractor Definition, with the issuance of the System Specifications. Modifications to the initial System Specifications that reflect approved redirections of the requirements are included in this step, even though they may occur later in the development process.

Tasks: Analyze system requirements

Analyze user's environment

Analyze computer program end product characteristics desired

Analyze similar or interfacing systems

Prepare system performance specifications

Obtain user's concurrence of system performance specifications

c. Information Processing Design--Step 3. Definition: Based on the design and performance requirements documentation from Information Processing Analysis, this step includes the definition of detailed design and performance requirements for functional elements of the information processing end product, e.g., translator, data retrieval and man-computer interaction. Information Processing Design results in concurred-upon and updated documents that detail the functions to be performed by the computer, computer program and interfacing operators.

AFSCM 375 Context: Information Processing Design occurs during Phase B, Contractor Definition. The resulting document, a firm definition of detailed functions, is equivalent to the "Contract End Item Detail Specification (Computer Program)--Part I." Modifications to the Part I Specifications that reflect approved design changes are included in this step, even though they may occur later in the development process.

Tasks: Interpret functional requirements (specifications) in terms of specific equipment, input types and volume, response time, operating environment

Produce a system flow diagram

Define system interfaces

Specify computations, logical manipulations, and transformations to be done within each functional area

Design the data base

Develop requirements for system data editing, formatting, storing, retrieving, and updating

Produce program system design documentation

Obtain user's concurrence on system design

Indoctrinate programming personnel

d. Computer Program Design--Step 4. Definition: This step is the determination of how the requirements established in the Information Processing Design (Step 3) will be implemented by a computer programmer. Included is the plan for actual data handling by the computer, and the computer program logic.

AFSCM 375 Context: Computer program design occurs during the Acquisition Phase, and contributes to the "Contract End Item Detail Specification (Computer Program)--Part II" that is produced as a part of Computer Program Coding and Checkout and is an input to the Preliminary Design Review (PDR). Computer program design modifications that reflect approved changes are included in this step, even though they may occur at a later date in the development process.

Tasks: Develop program test plans

Design logic and flow chart for each computer program component

Specify all input and output message formats

Coordinate design and communication requirements with executive control program requirements

Determine data rates and characteristics of input and output equipment

Design program files

e. Computer Program Coding and Checkout--Step 5. Definition: Based upon the detailed computer program design specification from Computer Program Design, this step includes all necessary work to produce and document the computer program in accordance with the current detailed design specification, and perform in-house tests. Included are such activities as coding, desk checking, computing tests (or runs), integration of individual units into a computer program system, preparation of the data base, logical and coding error detection and correction, compiling or assembling, and listing of

15

code. Computer Program Coding and Checkout results in a completed computer program end product, tested in-house to assure conformity with the current detailed specification, and ready for demonstration tests for the user and/or procuring agency.

AFSCM 375 Context: Computer Program Coding and Checkout occurs during the Acquisition Phase and includes Category I testing and evaluation of the computer program by the developer. This step results in a completed "Contract End Item Detail Specification (Computer Program)--Part II," which is an input to the Critical Design Review (CDR). Approved modifications to the computer program end product and to the Part II Specification are included in this step, even though they may occur later in the development process.

Tasks: Write coded program statements from detailed flow charts or other program design documentation

Desk check program code

Compile and check program code, and make necessary error corrections

Performance test individual programs

f. Computer Program Functional Test--Step 6. Definition: This step covers demonstration tests of the computer program end product conducted for the user and/or procuring organization, usually in a simulated environment at the developer's facility. Computer Program Functional Test includes conduct of the demonstration tests (based on test plans prepared as a part of Information Processing Design), analysis, and documentation of the test results. All necessary work to remedy errors or design deficiencies revealed by these tests should be charged to the appropriate previous steps, e.g., Information Processing Analysis, Information Processing Design, Computer Program Design, or Computer Program Coding and Checkout. If desired, costs for the original work in a step can be identified in a way to differentiate them from the costs of any similar work that is repeated as a result of testing. Computer Program Functional Test results in a computer program end product that is ready for demonstration tests in a live operational environment.

AFSCM 375 Context: Computer Program Functional Test occurs in the Acquisition Phase, and is equivalent to Category I Preliminary and Formal Qualification testing. Category I Formal Qualification testing is usually conducted at the facility designated for Category II testing.

16

Tasks: Plan functional test

         Prepare functional test benchmark problems

         Run functional test

         Make necessary coding or system design changes
         (charged to appropriate step)

   g.   Information Processing Integration Test--Step 7.   Definition: This
step covers demonstration tests conducted at an operational facility under
"live" environmental conditions and includes conduct of the tests (based upon
a test plan produced as a part of Information Processing Design), analysis,
and documentation of the results.   All necessary work to remedy errors
revealed by these tests should be charged to the appropriate previous steps,
e.g., Information Processing Analysis, Information Processing Design, Computer
Program Design, Computer Program Coding and Checkout, or Computer Program
Functional Test.   Information Processing Integration Test results in a
computer program that is a proven part of the information processing system
(end product), in conformance with the detailed design specifications.

AFSCM 375 Context: Information Processing Integration Test occurs in the
Acquisition Phase and is equivalent to Category II testing.

Tasks: Conduct test according to plan

         Analyze test results

         Initiate any modifications needed for computer programs

         Document test results

   h.   Information Processing Installation and Turnover--Step 8.   Definition:
This step covers all necessary work to install and check out the information
processing end product at the operational sites and will usually apply only
when there is more than one operational location.   This step also includes
user training as well as any phaseover activities, in the event that an
information processing system (manual or automatic) exists.   Information
Processing Installation and Implementation results in an operational
information processing end product at all sites.

AFSCM 375 Context: Information Processing Installation and Turnover occurs
in the Acquisition-Operational Overlap Phase, beginning with the installation
of the information processing contract end item at an operational site other
than the Category II site, and ending with turnover of the information
processing system to the user at the last operational site.

17

Tasks: Prepare user documentation

Advise user on data conversion (on in-house projects, the costs of
data conversion required to make the system operational are properly
included in this step)

Develop user training plan

Conduct training program

Conduct demonstration test

Assist in operational shakedown

i. Information Processing Program Maintenance--Step 9. Definition:
Information processing system maintenance is the process of improving,
changing, and correcting computer programs in an information system that is
currently operational. Program maintenance, including both revision and
error correction, is needed throughout the life of the information system.
Revisions are needed because operational requirements are continually changing
during both the development and operation of the system. Although operational
needs are projected during requirements analysis, in most cases they can be
neither totally defined nor totally implemented in the imposed time schedules.
Also, corrections must usually be made to the computer programs because errors
and operational deficiencies not detected in the routine testing of the programs
are usually discovered when the system becomes operational.

Much of the work of program maintenance personnel must be devoted to the
resolution of emergencies; a good share of the remainder, to modifications
required by hard-to-predict environmental changes.

AFSCM 375 Context: This step occurs during the Operational Phase of a system
life cycle. It includes incorporation of update and modification changes,
Category III testing, and phaseout of the System Project Office.

Tasks: Develop maintenance plan and organization

Establish communications between user and computer program developer

Establish internal communication channels

Establish change procedures

Process system changes

The above nine steps are consistent with the previous work at System
Development Corporation (1), with the exception of the addition of the first
and last steps. The feasibility analysis and system maintenance steps were

included to: (1) provide the means for collecting all of the costs associated with computer programming efforts, from the conception through program operation, and (2) provide a phase in which the formal planning needed for project management may be effected.

These nine steps are proposed as common structure for planning information processing system development. On some computer programming jobs and in some organizational environments, the distinction between all of these steps may not be practical in practice. However, if in adopting the cost-collection system an organization does decide to combine steps, the resulting combination should be applied uniformly for all programming jobs to assure comparable data from effort to effort.

2. <u>Planning and Reporting Work Breakdown Structure</u>. A primary requirement for a viable cost-collection system is ease and simplicity in the collection process, along with minimum interference with operational performance. This places any burden for the selection and reformatting of specific data items for control reports on administrative staff personnel; likewise, the restructuring of data to satisfy research objectives is the prerogative and responsibility of the individual researcher.

To guide the selection of data items for conducting research, the following general hypothesis was assumed: The cost of computer programming can be predicted from the characteristics of the programming job, including the requirements and estimates of certain product features, of the resources used, and the computer programming environment (3). Thus, the data such as personnel and computer characteristics are intended for analysis that would examine their impact on the recorded expenditure of resources. In addition to this general objective, specific research questions are raised in the following section; the purpose is to bring into sharper focus the reasons for collecting many of the proposed data items.

But for the worker engaged in research, it would be desirable to have a data base capable of supplying input to test various hypotheses framed in innumerable ways, since he simply may not know all of the directions that his research might take at the time he begins his project. A feasible way of providing this flexibility at a low cost is to record data as close to their source as possible, and to identify them with a hierarchy that describes the work breakdown structure for both product (end items) and work activities.

    a. <u>Work Breakdown Structure</u>. The work breakdown structure suggested below is based upon the division of information processing development into nine activities, or <u>steps</u>. The computer programming work on many information processing systems may be subdivided further into separate efforts or <u>computer program components</u>; this permits the assignment of a comparatively smaller team to each component for some of the step. The team is responsible

for the computer program design, coding, checkout, and functional test of these components. Likewise, each step may be composed of one or more activities, called tasks.

In many data automation projects, particularly smaller efforts, there may be no need to break the work into separate computer program components. For these projects, there may be only one computer program component, which is therefore identical to the information processing system.

The hierarchical arrangement of the information processing system, computer program component, step, and task is defined as follows:

```
INFORMATION PROCESSING SYSTEM

COMPUTER PROGRAM COMPONENT

STEP

TASK
```
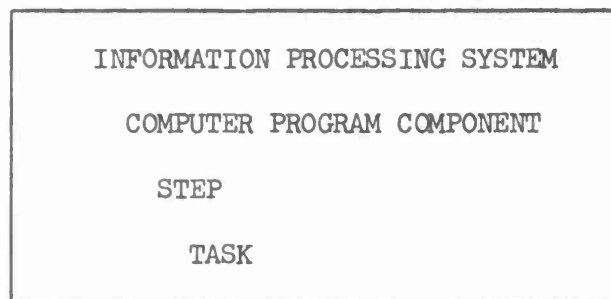
The lowest level of information processing system development at which a computer programming activity produces a computer program component is defined as the set of activities resulting in a product capable of operating in the computer as an entity. This concept is consistent with the definition of "Program Data Point" used in previous programming research at System Development Corporation (3).

The definition of the highest (system) level of information processing system development at which to collect data pertinent to any "total" system is more arbitrary. A computer program component to calculate payroll costs, for example, may be part of a larger personnel information system; this personnel information system in turn may be part of a management information system which, in turn, may be part of an industry forecasting model, etc. The pertinent question for cost reporting is, What part of a larger, or inter-facing, system's planning costs should be attributed to subsystems? Or, more directly, at what system-integration level should we stop identifying costs? A definition of an information processing system to delineate this upper bound is the following: For the purpose of cost collection and reporting, the Information Processing System category is the lowest level in a system hierarchy at which a management decision to proceed or not to proceed is, or has been, made. With this definition, the decision to proceed can be that based upon a cost-benefits analysis made in the feasibility study, Step 1. Then costs subsequently collected for this system level can be used to test the wisdom of the choice.

In the system for collecting and reporting costs all direct cost data--man hours by individual, computer time by hardware configuration, and elapsed time--must be identified with every defined task and every task is part of a step in the development process. Each step is identified with work on a particular computer program component which is part of the pertinent information processing system. This breakdown of tasks within steps and the steps themselves coupled with the earlier definition of computer program component and information processing system forms the basic structure against which costs are budgeted and subsequently collected in this system.

This collection system provides flexibility in that all four levels of reporting need not be used; but all levels are available if needed in specific instances. The minimum reporting recommended is the identification of cost data by step; these data could be analyzed to answer the most currently pressing questions about computer programming economics. Depending upon his needs for control, the manager responsible for development can collect more detailed costs by using additional breakdowns into computer program components, or steps into tasks.

    b. A Typical Work-Numbering System. To collect and identify source data within the hierarchy suggested above, an appropriate work order or charge-numbering system is convenient. The following seven-position code should prove satisfactory in most instances:

Typical Work Numbering for Cost Data Collection
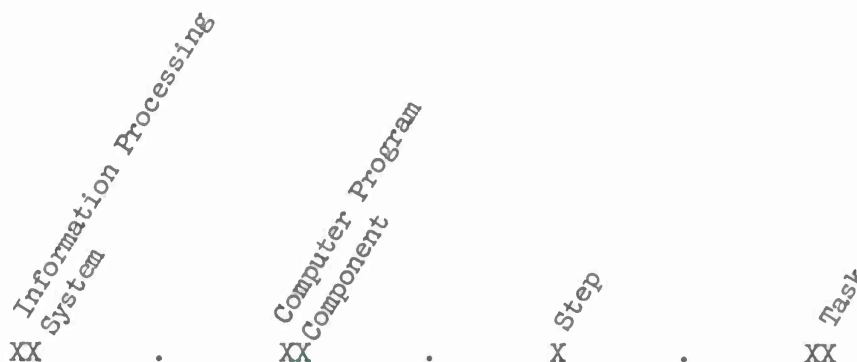
One position is needed for step identification. Obviously, the number of positions for the other entries required by any given installation would be a function of their own particular activities. Also, if different installations work on separate portions (components or steps) of a total information processing system, these installations should coordinate their numbering conventions.

Such a numbering system may be extended to identify costs with almost any kind of activity, characteristic, or output. For example, additional positions could be added to the typical work-numbering code described, to identify documentation efforts of various types, time spent in meetings or conferences or different types of efforts such as reprogramming.

c. Using the Work Breakdown Structure. Figure 1 shows how the work order numbers may be assigned to identify incurred costs to the appropriate category. In this example, parts of two separately planned and budgeted information processing systems are identified by numbers. The first information system is broken down into two components, perhaps to be worked on by different teams; the second involves no such component break-down. This organization creates a total of twelve cost categories for the first application, and nine cost categories for the second. Additional breakdown of steps into tasks was not elected in this example, although the possibility for such identification is shown by the remaining two digits (XX) that complete the seven-position code of the preceding section.

The collection of costs by step, the outstanding feature of work breakdown structure for the proposed reporting system, is illustrated in Figure 1. In this case, Steps 4, 5, and 6, Computer Program Design, Coding and Checkout and Functional Test, are repeated for each Computer Program Component; but resources expended on total system analysis, design, or integration cannot be allocated to components; instead, they are identified as part of the larger system's effort (note the "00" identity in the field assigned to computer program component). In some cases a computer program component may require work that may be appropriately charged to Step 1, 2, or 3 types of effort; where such work is clearly identified to a particular component, this may be readily recorded by using the component identifier.

3. Compatibility with Other Management Systems. This collection and reporting system for computer programming costs is a management information system for a specific application, computer programming. A recent survey by the Office of the Comptroller, Department of Defense, revealed over fifty different management systems in current use within the Department (15). The concurrent use of many systems in DOD may waste resources by duplication of effort; but a more serious problem is the potential for confusion, and the saturation of working organizations with various reporting requirements, to the detriment of their productive functions. The cost of collecting and reporting data is an important consideration in striving for efficient Department of Defense operations. Particularly important is the high cost of satisfying the reporting requirements imposed on contractors by DOD; one study (16) estimated the Fiscal Year 1963 contractor reporting cost to be between $250 million and $400 million.

Since the cost-collection system for computer programming described in this report is intended to have general applicability, aspects of its compatibility with several important existing government management reporting systems are

Data Identification Category	Typical Work Order Number
First Information Processing System	
Step 1	01.00.1.xx
Step 2	01.00.2.xx
Step 3	01.00.3.xx
First Computer Program Component	
Step 4	01.01.4.xx
Step 5	01.01.5.xx
Step 6	01.01.6.xx
Second Computer Program Component	
Step 4	01.02.4.xx
Step 5	01.02.5.xx
Step 6	01.02.6.xx
Step 7	01.00.7.xx
Step 8	01.00.8.xx
Step 9	01.00.9.xx
Second Information Processing System	
Step 1	02.00.1.xx
Step 2	02.00.2.xx
Step 3	02.00.3.xx
Step 4	02.00.4.xx
Step 5	02.00.5.xx
Step 6	02.00.6.xx
Step 7	02.00.7.xx
Step 8	02.00.8.xx
Step 9	02.00.9.xx

FIGURE 1

EXAMPLE OF WORK ORDER NUMBER ASSIGNMENT

briefly reviewed here. To be truly compatible, the system proposed here would require tailoring before it was integrated into another working system; such tailoring may undermine the uniformity needed for universal application. Also, the system would change over time because of changes made to present systems. But it is important that the method for data collection proposed should not conflict with any existing procedures, since such conflicts would compound the problems of gaining acceptance of programming cost data collection within operating organizations. Also, the existing systems and procedures, such as the AFSCM 375 series, were intended to provide needed tools for management in the Federal Government; it would be a significant contribution if the material proposed herein helped to extend these principles into the area of computer programming, even if some modifications and interpretations were later required.

To deal with the compatibility question in data-collection systems, several dimensions should be considered. The principal ones are work breakdown structure, data element definition, level of aggregation, time of collection, and data format. Work breakdown structure, the set of standard cost accounts into which data are segregated, produced major problems during the implementation of PERT/Cost systems; the traditional chart of accounts used for financial reporting simply did not have provisions for collecting and assembling cost data identified with a project or activity. Data element definition, including the level of aggregation of the data represented, is a matter of particular concern here; personnel using the system must know precisely what they are dealing with, and whether the data requested are the same as those requested or available in other reports for other management systems. The time period between reports may vary for different reporting systems; these differences should be recognized and whenever possible made the same. Finally, uniform data formats for different systems can help assure easy handling and recognition of data.

    a. The Planning Program Budgeting System (PPBS). The Department of Defense program planning system, as defined by DOD Directive 7045.1, has become an important part of the Department's environment. By Presidential order, as of August 1965, the Program Budget concept is being introduced to other areas of the Federal Government (17). Although many projects (programs) will fall below the present thresholds (a change of $10 million for systems in R&D, a $25 million total program, or any change in obligational authority), we may assume that all programs within Air Force Systems Command, Department of Defense, and eventually the Government, will be affected by the management information and control concepts inherent in the Program Budget (i.e., comparable structuring of plans and programs according to resources, uses, and implementation).

Within the Program Budget context, the Office of the Secretary of Defense (Comptroller) has been given responsibility for designing a Selected Acquisitions and Information Management System, SAIMS, one part of which

consists of Cost Information Reports (CIR), and the other a cost and schedule performance system. At present, CIR provides a comparable cost-reporting structure intended for aircraft, missile, and space systems.

The major compatibility issue concerning PPBS or CIR is whether or not the proposed cost-collection system for computer programming can operate within the required work breakdown structure of the former. The PPBS structure spans seven levels of cost information, each of which is a more detailed breakdown of the preceding level (18). The reporting detail in Levels 1 through 6 is mandatory for all Department of Defense organizations. Within this system, Level 5 is the highest level at which computer programming is cited as a separate item in any of the referred Department of Defense agency documents. The standard breakdown in this case (19) is:

Level 4	Level 5	Level 6
Prime Mission Equipment	Computer Programming	RDT&E Program Acquisition Utility Maintenance and Operational Exercises Other

In this case, Level 6 could include an information processing system as defined earlier and Level 7 could include a computer program component. For command and control systems, computer programming, when it is not a part of a Prime Mission Equipment, is usually broken out at a Level 6 item, or under such Level 5 breakdowns as Technical Manuals and Orders, Engineering Data, and Program Management Data. A similar modification could be made in these cases.

Elsewhere in the development of command and control systems, or, for that matter, missile, aircraft, or space systems, whatever computer programming that is done as a part of other activities, e.g., system analysis, system evaluation, program management is subsumed under other categories.

Since computer programming is handled in a variety of ways in PPBS, the computer programming cost-collection system proposed in this report is not incompatible with existing PPBS structure. The question as to whether computer programming could be handled in a uniform way in PPBS has not been answered yet. However, when standards do exist, such as industry standards for functional cost categories (e.g., labor, travel) at Level 6 and below, they would need to be modified to accommodate this system to reflect the particular needs of the computer programming process.

   b. AFSCM 375 Series. This series of management procedures was developed to improve the technical as well as economic management of the procurement cycle for military weapons systems. The cost-collection system for computer

programming proposed here is compatible with these procedures. The specific relationship of the steps established for the computer programming process and the phases of the AFSCM 375 procurement cycle were outlined earlier in this section.

c. PERT/Cost. A major problem encountered in the implementation of PERT/Cost management systems was the orientation of the existing accounting systems into departmental cost breakdowns, and into the traditional accounting categories such as accounts receivable and inventories. PERT/Cost, on the other hand, required planning and subsequent cost collection by project, subproject, and activities within these directed efforts. Since projects typically cross departmental lines, and activities on projects bear only indirect relationships to the corporate balance sheet and income statement model, existing accounting structures were not adequate for PERT/Cost reporting.

The system proposed here for collecting programming costs is very similar to the PERT/Cost model, but the planning structure does not identify specific dependencies among events that a PERT diagram would reveal. Costs are collected by effort, i.e., project, and by steps within projects whose completion would constitute major milestones in a PERT network. Likewise, the completion of tasks within steps can, and should, be planned as finite events. The comparison of estimated versus actuals, although the mechanics, e.g., the forms, differ, is equivalent to the PERT/Cost process.

d. AFR 300 Series. Air Force Regulation 300-3 (4) prescribes procedures and responsibilities for the design, implementation, modification, and maintenance of management supporting data systems. It implements the objectives and policies outlined in a companion document, AFR 300-2, "Data Automation Objectives and Policies." The principal contributions of AFR 300-3 are the establishment of the Data Automation Proposal (DAP) procedures for identifying and submitting proposals for new data automation applications, and the Data Project Directive (DPD), which provides the charter for command or agency initiation of a system development project and establishes the scope and parameters of the developmental effort.

AFR 300-3 is related to this proposed cost-collection and -reporting system in several ways. AFR 300-3 establishes the requirement for Data System Designation codes for identifying approved systems that are under development; these designators are equivalent to the Information Processing System identifiers discussed herein. Three generalized tasks, or steps, are singled out in AFR 300-3 that have counterparts in this system:

AFR 300-3	Suggested Cost-Reporting System
Data System Analysis	Information Processing Analysis (Step 2)
Data System Design	Information Processing Design (Step 3)
Data System Programming	Computer Program Design, Code, and Functional Test (combined Steps 4, 5 and 6)

AFR 300-3 also requires data systems specifications and workload descriptors, technical characteristics of ADP equipment, and programming languages to be used. Those items are the same as many of the other items of data requested for research in this report. Thus, to a considerable extent the proposed cost-collection and -reporting system provides a device for implementing many of the existing requirements spelled out in AFR 300-3; and in addition, the means are established for determining and evaluating data automation progress periodically as the work proceeds.

SECTION IV

DATA COLLECTED AND REPORTED

This section is divided into three parts: (1) Source Data Collected from Operating Personnel; (2) Reports for Management Control; (3) Data for Research. Each part includes a set of reporting or collection forms. These forms could be used as presented or modified according to the needs of the adopting organization. The forms and the data they contain are listed below:

. Source Data Collected from Operating Personnel

The Weekly Activity Report (Figure 2)--contains the direct time spent on a task or step by all personnel assigned to the development effort.

. Reports for Management Control

The Information Processing System Status Report consisting of the following three forms:

The Summary Weekly Activity Report (Figure 3)--contains a summary of the data collected by the Weekly Activity Report.

Information Processing System Progress and Costs (Figure 4)-- contains data on direct costs, man-months and computer hours, budgeted, expended to date, and estimated for remainder of development effort, by steps, tasks and milestones.

Information Processing System Progress Graphs (Figure 5)-- contains graphical representation of budgeted, expended to date and estimated for remainder of development effort, of direct costs, direct and indirect costs and number of object instructions.

The Information Processing System Management Summary:

The Information Processing System Management Summary (Figure 6)-- contains data on direct and indirect costs for all computer program components within the information processing system.

. Data for Research

Information Processing System Requirements Summary (Figure 7)-- describes the required characteristics of the information processing system and its computer program components.

Production Computer System Configuration (Figure 8)--describes the hardware configuration(s) used in program development.

Personnel Data Characteristics (Figure 9)--describes the personnel assigned to work on the development of the information processing system.

Computer Program Component Characteristics Summary (Figure 10)-- describes the salient characteristics of a computer program component within the information processing system.

Development Environment Summary (Figure 11)--contains a description of the environment in which the information processing system and its components were developed.

The Development History (Figure 12)--the collection of all forms into a detailed, comprehensive, and descriptive picture of the total development effort.

Within any organization adopting this system, changes may be made in both the structure and content of these forms. For example most organizations already have some cost reporting requirements and may use forms similar to some proposed here, particularly the Weekly Activity Report. In such cases, the existing forms of the utilizing agency could be changed to include spaces for recording the information proposed for collection in this system. On the other hand, several of the forms illustrated in this report (e.g., the Information Processing System Progress and Costs) may not correspond to any existing reporting format in an organization using the system; then the illustrative forms may be used as a reference for both format and content.

This system is also quite flexible; it can be used to collect data to meet either or both of its major objectives--management control and research. For example, if the system is to be adopted in an installation that is primarily interested in the control aspect and not in conducting research, only one basic collection form is required: the Weekly Activity Report and the forms and data for research could be ignored. In other cases managers may have specific questions whose answers could provide improved planning factors. Then, additions, deletions, or modifications may be made to the data collected as inputs for analysis.

1. Source Data from Operating Personnel. The Weekly Activity Report (Figure 2) is the primary form for source data collection in the system. It is to be completed by each person who charges directly to the project: system analysts, programmers and other personnel, including secretarial and documentation support. The primary intent is to collect direct labor costs by a unique charge code, such as a work order number in the system described earlier. Later, these man hours may be reconciled to the products by using the charge code. The

charge code uniquely identifies the information processing system, the computer program component, the programming process step, and tasks, within each step, and allows for additional breakdowns that may be used at the discretion of the project manager. In addition to the man hours that are collected, the form provides for the reporting of progress toward the completion of scheduled tasks.

The Weekly Activity Report should be completed each week, checked for accuracy by a supervisor, and sent to the staff personnel primarily responsible for operating the cost-collection system. Since the Weekly Activity Report supplies the raw data for both management control and research purposes, these data should be as error-free as possible. To help assure accuracy, personnel should complete the appropriate column daily.

The following information is compiled by the Weekly Activity Report (the numbers correspond to the numbered blocks in Figure 2):

1. The Sunday date--the date of the Sunday ending the work week for which the data are being collected.

2. Employee name.

3. Employee number.

4. Organization staff--the name of the organization staff responsible for the development of the Information Processing System or its parts (i.e., Computer Program Component) to which the employee is assigned.

5. Charge code--consists of five subgroups:

    Information Processing System (IPS)

    Computer Program Component (CPC)

    Programming process step

    Task

    Detailed breakdown--provision is allowed for a detailed breakdown of the task, or cost identification to other activities (e.g., documentation meetings, etc.), at the discretion of the project manager

6. The total charge--the total hours expended on each step or task, i.e., work order number, for this week.

31

WEEKLY ACTIVITY REPORT							1. Sunday Date	
2. Employee Name							3. Employee Number	
4. Organization Staff								

Distribution of Hours Worked by Charge Code								
5. Charge Code IPS / CPC / STEP / TASK / OTHER	MON	TUE	WEDS	THURS	FRI	SAT	SUN	6. Total by Charge Code
7. Total Hours Worked								
8. Absence: Vacation								
Sickness								
Other Absence								

9. Progress Report
10. Special Problems

FIGURE 2

SAMPLE WEEKLY ACTIVITY REPORT

7. Total hours--the sum of hours worked by day, extended to agree with the total, by charge code, for this week.

8. Absences--allowance is made here to charge time for vacation, sickness, or other absences.

9. Progress report--employee should enter indications of progress toward the completion of scheduled tasks, e.g., number of instructions written, documents completed or partially completed.

10. Special problems--provision is made for the reporting of problems that affect the progress of work such as machine failure, system change, travel, etc.

In designing this form, it was assumed that some secondary sources of information would be available in supplying data on the number of computer runs made by each programmer and the computer time used in each run. If such data are not readily available, computer time and number of runs should also be collected from operating personnel on forms similar to the one shown in Figure 2.

2. Reports for Management Control. Two reports for management are proposed as aids for project control: the Information Processing System Status Report, and the Information Processing System Management Summary. Each report is intended for a different level of management, and each provides timely data to inform management of the progress of a programming effort and the expenditures of the resources in the effort.

The frequency of compiling these reports is determined by the recipient managers, who specify the intervals at which they want to receive the reporting forms.

The basic data contained in these management reports are also intended for use in research; however, these data are organized and presented in the management reports to be most meaningful for management action. Thus, the presentation of these data should be slanted toward the particular needs of the management of each organization using the system without regard to the needs for uniformity in data to be entered into the data bank. Therefore, depending upon the needs and tastes of the managers involved, these management summaries may differ from one organization to another. For example, one of the forms includes graphs to compare actual expenditures with estimates; these graphs could be replaced or accompanied by tabular data according to the preference of the reviewing manager.

a. The Information Processing System Status Report. The Information Processing Status Report is received by the manager responsible for the entire development effort. This document consists of the following three forms:

. The Summary Weekly Activity Report (Figure 3)

. The Information Processing System Progress and Costs (Figure 4)

. The Information Processing System Progress Charts (Figure 5)

(1) The Summary Weekly Activity Report. The Summary Weekly Activity
Report (Figure 3), as the name implies, summarizes the data reported on the
Weekly Activity Report (Figure 2) and adds computer usage data collected from
other sources such as a log of computer use automatically recorded and printed
by the computer. This report lists the personnel involved in the productive
effort during the reporting period along with the time spent and computer time
and runs by individuals. This information, if accurately reported, aids in
determining if the assigned personnel are in fact devoting the time to the
effort as originally proposed. Examination of these reports can also reveal
if personnel continuity is being maintained to expected levels.

The Summary of Weekly Activity compiles data for man-hours, number of computer
runs, and the amount of computer time used by persons charging to the particu-
lar effort. The data are further ordered by charge code, thus creating a
summary by charge code and employee. The man-hour content parallels that found
in the Weekly Activity Report. Usually the data for computer time and number
of computer runs can be supplied by the organization responsible for operating
the computer facility used in the development. If such data are not provided,
some provision should be made for their collection. As suggested earlier one
way would be to modify the Weekly Activity Report to request these data.

Large efforts may require several of these forms to include all of the personnel
involved. The Summary provides a convenient device to organize data regarding
expenditure of resources. Space is also provided to summarize work accomplish-
ments. Thus one form provides a reasonably complete picture of development
expenditures and progress.

The following information is contained in the Summary of Weekly Activity
Report (the numbers correspond to the numbered blocks on the form shown in
Figure 3):

1. Organization Staff--name of organization staff responsible for the
    development of this information processing system (or portion
    thereof, i.e., computer program component).

2. Sunday Date--the date of the Sunday ending the work week for
    which the data are being summarized.

3. Employee Number--enter employee number at the top of each 3-part
    column. In these columns the data are recorded for hours worked,
    number of computer runs, and computer time (in minutes) used for
    each employee.

34

SUMMARY OF WEEKLY ACTIVITY		1. Organization Staff		2. Sunday Date	

Distribution of Hours Worked, Computer Runs and Computer Time by Charge Code for Each Employee

4. Charge Code IPS / CPC / STEP / TASK / OTHER	3. Employee Number			Employee Number			Employee Number			Employee Number			Employee Number			6. Total by Charge Code		
	Man Hours	R U N S	Comp Time	Man Hours	R U N S	Comp Time	Man Hours	R U N S	Comp Time	Man Hours	R U N S	Comp Time	Man Hours	R U N S	Comp Time	Man Hours	# of Runs	Comp Time
5. Total by Employee																7.		
8. Absence: Vacation																9.		
Sickness																		
Other Absence																		

10. Associated Progress Report

11. Special Problems

FIGURE 3

SAMPLE SUMMARY WEEKLY ACTIVITY REPORT

4. Charge Code--consists of five subgroups.

    a. Information Processing System (IPS)

    b. Computer Program Component (CPC)

    c. Programming process step

    d. Task

    e. Detailed breakdown--provision is allowed for a further break-down of the task at the discretion of management.

5. Total by Employee-sum of the data for each employee in each of the three columns.

6. Total by Charge Code--sum of the data for each charge code in each of the three columns.

7. Summary Total--the summation of the totals for each employee extended to agree with the total, by charge code, in each of the three columns.

8. Absences--the number of hours charged to absence for each employee.

9. Total Absence--summation of the total absence for all employees on the form.

10. Associated Progress Report--a summary of the progress reports from the individual Weekly Activity Reports.

11. Special Problems--a summary of the problems from the individual Weekly Activity Reports.

(2) The Information Processing System Progress and Costs (Figure 4). Information Processing System Progress and Costs Form is used to monitor the work flow and total direct and indirect resources expended. This form is completed by staff personnel, and its purpose is management control; that is, comprehensive comparisons between estimated and actual costs and schedules are periodically (weekly or monthly, depending on the needs or desires of management) presented for examination.

By studying the Progress and Costs summary, management can determine whether the work is proceeding within the planned boundaries set for the development or if some corrective action is required.

The Information Processing System Progress and Costs form contains the following data:

			INFORMATION PROCESSING SYSTEM PROGRESS AND COSTS							

1. Information Processing System:					2. From: To:					

3. Step	4. Milestone (Task Completion) Within Step	5. Milestone Completion Date		Direct Costs						
				6. Original Estimate		7. Actual to Date		8. Estimate to Completion		
		Est.	Act.	a. Man Hrs	b. Comp Hrs	a. Man Hrs	b. Comp Hrs	a. Man Hrs	b. Comp Hrs	
Info Proc. Feasibility Analysis										
Info Proc. Analysis										
Info Proc. Design										
Comp Prog. Design										
Comp Prog. Code and Checkout										
Comp Prog. Functional Test										
Info Proc. Integration Test										
Info Proc. Installation and Turnover										
Info Proc. Program Maintenance										
9-12. Total Direct Cost ($) G & A ($) Profit or Fee ($) Total Price ($)				9a. 10a. 11a. 12a.		9b. 10b. 11b. 12b.		9c. 10c. 11c. 12c.		
13. Total Number of Delivered Object Instructions				13a.		13b.		13c.		

FIGURE 4

SAMPLE INFORMATION PROCESSING SYSTEM PROGRESS AND COSTS

37

1. The Information Processing System--the name of the Information Processing System being developed.

2. Reporting Period--the beginning and end dates for the time covered by data in this form.

3. Step--the programming process steps within which management control is critical to the smooth flow of work and resources, as defined in Section III.

4. Milestone--a definable point in development within a process step; used as progress markers for project development. The milestones for a process step should be determined by the lead programmer and/or his supervisor and are characterized by an event at a definite point in time, i.e., a completion of a specific task.

5. Milestone Completion Date--the estimated date of completion of a milestone as determined during the planning phase, and the actual completion date for this milestone.

6a. Estimated Man Hours--the number of man hours initially estimated for the indicated milestone.

6b. Estimated Computer Hours--the number of computer hours initially estimated for the indicated milestone.

7a. Actual Man Hours--the actual number of man hours expended to date on the milestone for which the estimate in 5a was made.

7b. Actual Computer Hours--the actual number of computer hours used to date on the milestone for which the estimate in 6b was made.

8a. Man Hours Estimated to Completion--the number of computer hours currently estimated to complete the milestone referred to in 5b and 6b.

9. Total Direct Dollar Cost Charged Directly to the Project--(a) the originally estimated total information processing system development cost; (b) the actual direct dollar costs expended to date; (c) total direct dollar costs currently estimated for completion.

10. Total Indirect Dollar Costs, e.g., overhead, burden, etc.--(a) total originally estimated development cost; (b) actual indirect dollar costs expended to date; (c) indirect dollar costs currently estimated to completion.

11. Total Profit or Fee Dollars--(a) total originally estimated profit or fee for the development effort; (b) actual profit or fee allocated to date, (c) current estimate for completion.

12. Total Price; the sum of 8, 9, and 10--(a) originally estimated for Information Processing System development; (b) actual expenditures to date; (c) currently estimated for completion.

13. Number of Object Instructions to be Delivered--(a) number of instructions originally estimated for the Information Processing System; (b) actual number of object instructions completed to date; (c) estimated number of object instructions to completion.

(3) The Information Processing System Progress Graph. The Information Processing System Progress Graph is a collection of four graphs; three graphs depict the actual versus estimated flow of project resources and the fourth is a pictorial presentation of the estimated against completed number of object instructions to date. (In organizations that use standards for documentation and in which some experience has accumulated in estimating the amount of documentation and its growth, a graph of estimated versus actual number of pages could also be used.)

This form is maintained by staff personnel, and its main function is to give management a comprehensive graphical representation of development progress. Direct man hours, direct computer hours and total dollar costs (direct and indirect) are the three resources graphically presented in this form. The hypothetical illustration in Figure 5 should be self-explanatory.

b. The Information Processing System Management Summary (Figure 6).
This Summary is a comprehensive form containing information on the direct and indirect costs that have been charged to the development of the Information Processing System and its computer program components. Since this form is intended for higher levels of corporate management, the recipient probably will desire to have the information on this report combined with that on efforts other than computer programming, e.g., costs of marketing or production in a large corporation or costs of other developments (subsystems in weapons or electronic systems) in an Air Force System Project Office; this combination would permit an evaluation of the resources allocated to computer programming in the broader context of the organization's ultimate objectives.

In addition to the accumulated costs for each Computer Program Component in the information processing system, the Management Summary also contains data on the original cost estimates and expected costs to completion for each program component. Such information may be useful in reevaluating resource commitments.

39

INFORMATION PROCESSING SYSTEM PROGRESS GRAPHS

Date:

Direct Man Hours

t_o

Direct Computer Hours

t_o

Total $ Spent (Direct + Indirect)

t_o

Object Instructions

t_o

LEGEND:
t_o = Current Date

Actual _____

Original Estimate . . . .

Estimated From t_o _ _ _ _ _ _

Name of Information Processing System:
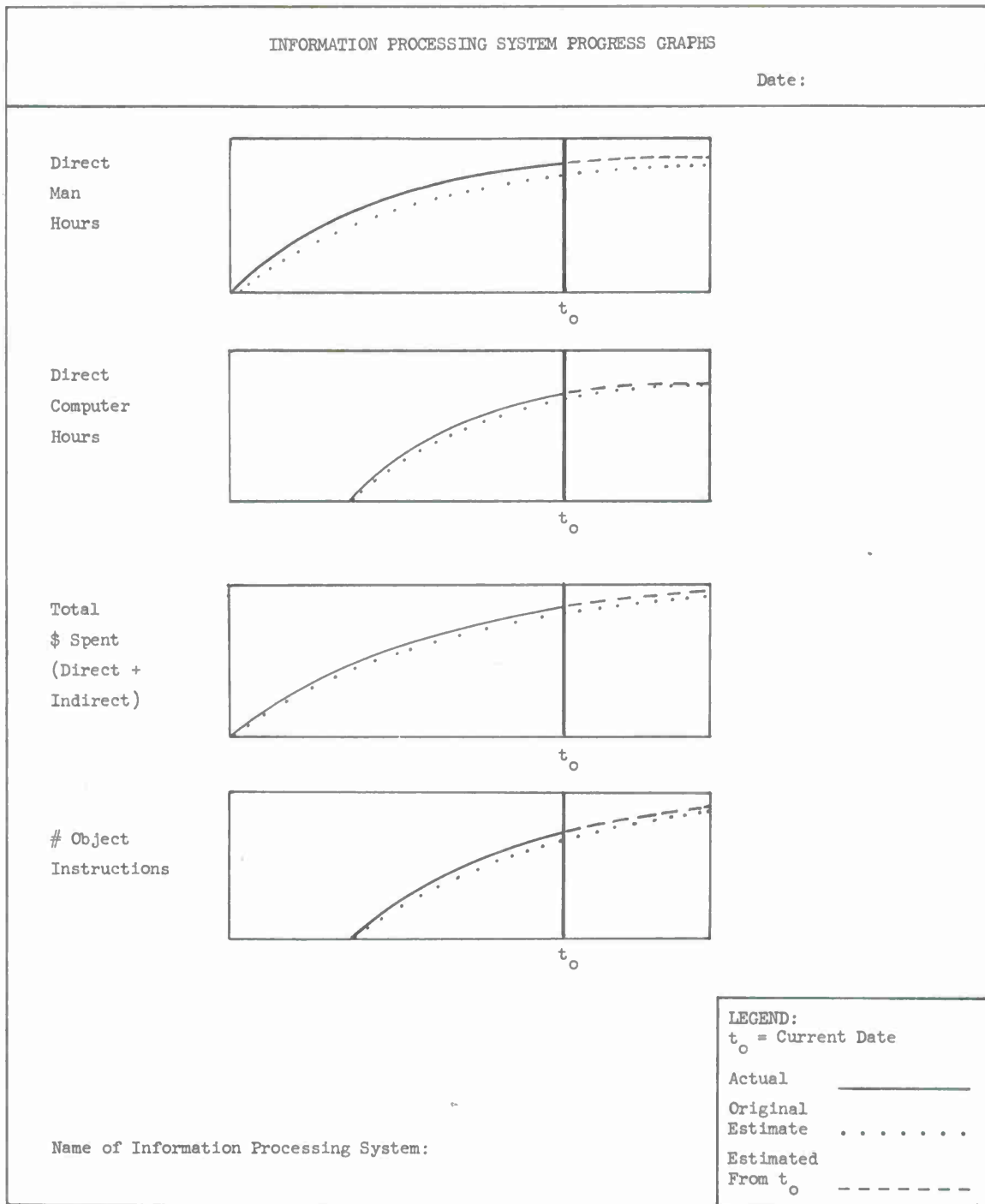
FIGURE 5

SAMPLE IPS PROGRESS GRAPHS

| | | INFORMATION PROCESSING SYSTEM | | | | | | | | | | |
| | | MANAGEMENT SUMMARY | | | | | | | | | | |

Note: MH = Direct man hours

      CH = Direct computer hours

   TTL $ = All costs, direct and indirect

Period Covered

From: _____ To: _____

Information Processing System	Computer Program Component	Original Estimate to Date			Actual to Date			Estimate to Completion			Remarks
		MH	CH	TTL $	MH	CH	TTL $	MH	CH	TTL $	
TOTAL											

Remarks:

FIGURE 6

SAMPLE INFORMATION PROCESSING SYSTEM MANAGEMENT SUMMARY

With the above collection of forms, both the middle (the Program Component) and higher (corporate) management are constantly kept in touch with the development and progress of the components in a system, and are thereby better able to identify and correct any difficulties that arise during the development of the system.

3. Data for Research

a. Discussion. To collect data on cost factors--the probable influences on cost, five additional forms are included in the collection system. These data are intended for use in analyses to derive improved cost estimating relationships. The five forms contain data items that the Programming Management Project (PMP) has previously identified (3) as having an effect on computer programming costs.

Nearly one hundred such factors were identified for a combination of three of the nine process steps defined earlier in Section III. The three steps for which the factors were identified were: Computer Program Design, Computer Program Coding and Computer Program Checkout. Many of these factors are believed to influence costs for the entire development of an information processing system as well as individual steps among the other six.

In the earlier PMP work the cost factors were grouped into five logical categories that influence computer program development. These categories correspond to the forms used in the cost-collection system.

(1) Operational Requirements

(2) Data Processing Equipment

(3) Programming Personnel

(4) Program Design and Production

(5) Development Environment & Management Procedures

Extensive data collection from various military agencies and industrial organizations and iterative statistical analyses (3, 11, 12, 13) were performed to determine those factors that had the strongest effect on the cost of computer program development. The items in the collection forms contained in this report reflect the results of these analyses. That is, those factors that have had a statistically demonstrable affect on the cost of program development are proposed for collection.

In addition to the items derived from previous analytical work, additional items were added to the collection forms to address some of the many questions being asked by the electronic data processing community concerning the costs

42

of computer programming. Many of these questions, listed below, remain unanswered, due mainly to the lack of reliable data with which to investigate the factors that influence product cost and quality.

To conduct the cost analyses, the cost data in two forms described earlier, the Weekly Activity Report and the Summary Weekly Activity Reports (Figures 2 and 3), must be combined with the data for research. But these two forms alone provide data for profitable analyses. For example, the basic cost data for both manpower and dollars could be analyzed to determine the percentage of costs that are being expended in each of the appropriate steps of the information processing system presently under development. This type of analytical result could be used to establish a range of values based upon a large number of systems, and subsequently those "norms" could be used to identify large deviations. Further, using the research data, such deviations could be related to specific characteristics of the information processing system or the tools used to develop them, thereby creating a factor to use in planning future efforts of this same type.

Although the data recommended for collection as inputs for analyses include a large number of items, the list presented here will by no means be exhaustive. The collection of specific items of data depends upon the intent and/or interest of the particular organization that adopts the system. Most of the items requested are intended as inputs to conduct statistical analyses similar to those conducted earlier by the Programming Management Project at SDC. However, analyses conducted with those new data could be aimed at providing newer planning factors and at establishing equations for each step in the development process as well as the entire process, as contrasted with the earlier work to derive such guidelines for the aggregation of computer program design coding and characteristics.

A specific organization using this system may eliminate a large portion of the recommended data items, either because they have no interest in providing answers to specific questions that require these data to derive answers, or because they do not have a sufficient number of projects under way to supply adequate amounts of data for meaningful statistical analyses.

For example, a large organization such as the Air Force might adopt a subset of the proposed data items as a reporting requirement for both in-house and contractor organizations involved in computer program development. In this case because many of the development efforts are under way in the Air Force, large amounts of uniform and reliable data would be collected to answer some of the basic questions. In addition, if the adopting organization pursues various kinds of developmental efforts, the collected data may be partitioned according to some organized scheme, e.g., division into representative types of applications present--business programs, scientific programs, utility and support, etc. The purpose of such divisions would be to search for classes of more homogeneous data, with smaller spreads in cost than the entire collection. If these classes were found, the derived cost estimating relationships would be more precise.

The data items to be collected are intended to help the economics of computer programming answer specific questions on such as the following:

1. What is the relationship of machine time cost versus computer program development cost?

2. What is the relationship of machine speed to the cost of machine time used in computer program development?

3. What is the relationship of machine speed to total program development costs?

4. What percent of total costs and elapsed time occur during each step in the computer program development process? Are total costs including maintenance reduced by investing more resources in certain steps?

5. What is the effect of program size on cost per instruction?

6. What is the relationship between elapsed time for program production and production rate?

7. What is the relationship between elapsed time for program production and the number of programmers assigned to a project?

8. What is the relationship between the number of programmers and/or analysts assigned to a project and the production rate?

9. What is the relationship between computer turnaround time and production rate?

10. What is the relationship between programming costs and programmer experience and other personnel characteristics?

11. What is the relationship between program development cost and the participation of programmers in system and program design?

12. What is the relationship between documentation and programming costs?

13. What is the cost relationship between a program produced by means of a compiler and a program produced by means of an assembler (symbolic code)?

14. How do the size, nature and availability of the data base affect program development costs?

15. What effect does required response time specified have on programming costs?

16. What effect does the number of user organizations have on programming costs?

17. What are the differences in computer programming costs with various methods of computer operation--open shop, closed shop, and multi-access (interactive) on-line.

The following paragraphs identify and define the items of data needed as inputs to analyses to answer the questions listed above. As mentioned earlier, these data items are grouped into classes such as systems requirements, personnel, hardware used for development, etc. The items in each class are shown on a typical form that could be used to record the values. Finally, a way to assemble all the completed forms into a Development History as an entry into a data bank could be useful.

b. System Requirements Summary. The data items requested in Figure 7, System Requirements Summary, are salient characteristics of the information processing system. Items on this form may refer to both the total information processing system and its component computer program. This form is to be completed for each development effort that corresponds to a component or set of components of the information processing system. Whenever possible the system requirements should be identified with a specific computer program component as well as the total system. If these components have different requirements that are known, individual forms would be completed for each. For example, assume that an information processing system containing several components has a total data base requirement of 10,000 words. Some portion of this data base is used by each component in the system, and at times it is possible to identify and record this portion.

During the information processing analysis and design steps, the requirements usually become stable enough for the responsible staff man to be able to complete this form, using the documentation output from the analysis and design steps as the source information.

Despite thorough analyses of requirements and correspondingly thorough design efforts for the information processing system, requirements tend to change during the development cycle of the system. The staff personnel who were responsible for the original should record any such changes, using updates, as addenda to be attached to the original form. These requirements updates provide a record of the extent and frequency of changes during development.

The Information Processing System Requirements Summary consists of the following items:

INFORMATION PROCESSING SYSTEM COMPUTER PROGRAM COMPONENT/REQUIREMENTS SUMMARY	DATE PREPARED

1. Information Processing System Name	2. Computer Program Component(s) Name

3. Required Hardware	4. Availability of Required Hardware

5. Identify Compiler/Assembler	6. Availability of Compiler/ Assembler

7. Completeness of Requirements Documents

8. Interface Requirements

9. Innovations Required

10. Response Constraints

11. Number of Words in Data Base: Total System - These (This) Components -	12. Number of Classes of Items: Total System - These (This) Components

13. List all Computer Program Components in System:

14. Special Tools Required:

FIGURE 7

SAMPLE SYSTEMS REQUIREMENTS SUMMARY

1. Information Processing System--name of the information processing system of which the computer program component(s) is a part.

2. Computer Program Component(s)--name or give identification number of the computer program component(s) whose requirements are being summarized.

3. Indicate all computers and peripherals required for system development that are specified in the system requirements.

4. Identify all of the hardware components specified in Item 3 above, which are not available and require development activities (including installation and testing such as acceptance test).

5. Indicate which compiler(s) and/or assembler(s) are to be used in program development.

6. Indicate whether the compiler(s) and/or assembler(s) indicated in Item 5 is currently available (tested or certified in some way, e.g., successful runs of benchmark problems) for use in the development effort.

7. State whether the operational requirements are defined and documented, both for the total system and for each computer program components prior to start of computer program component design. If not comment on deficiencies.

8. Describe all interface requirements between this information processing system and other systems and/or programs.

9. Indicate those features of the information processing system that require innovation in design and production, and specify, if possible, the corresponding computer program component in which this innovation had to be considered. Innovation is defined as a significant portion of the design and production of the system and/or its components, that involved an application or some programming techniques that were new to the personnel assigned to these tasks.

10. Indicate the "response time to query" constraints imposed on the system. "Response time to query" constraint is specified minimum time delay between input query and the required output response.

11. Include the number of words in the data base for the total information processing system and the computer program components.

12. Give the number of classes of <u>items</u> in the data base for the total information processing system and that portion allocated to the computer program component(s). Classes of items are categories such as name, social security number, age, salary, or any other characteristic of information which describes something, e.g., a person, a ship, etc., for which there are many entries. An item is equivalent to a field, on a data processing format.

13. List all separate computer programs that are contained in this information processing system.

14. State if special programming tools, such as debug and test aids, are required for development, both for the total system and for a computer program component. Indicate the availability of any of these needed tools.

c. <u>Production Computer System Configuration</u>. In the development of a computer program system, several computer hardware configurations may be used during various stages of program production. This form (Figure 8) is designed to detail these configurations.

In organizations with one or two computer installations that are relatively stable from year to year, the configurations may be described elsewhere and a reference noted on the Computer Configuration Form. If, however, the creation of the hardware system was a direct result of the specifications indicated in the Systems Requirements (Figure 7), the configuration(s) should be briefly described. One of these configurations will be the computer configuration in which the computer program component or information processing system will actually operate even if other configurations are used in the development effort.

If the same computer system (<u>or</u> a physically different system with the same configuration and identical labels such as model number) was used during all stages of the program development, only one configuration need be indicated. However, if hardware systems of different configurations are used, each one should be noted by supplying the following data on <u>each</u> configuration, if different computers are used for different computer program components in the information processing system.

1. Information Processing System--name or identification number of the information processing system. If different computer program components are developed on different configurations indicate the component label here and complete an extra form for each component.

2. Open Shop--an installation where the programmers have "hands on," direct access to the computer for compiling, debugging, and operation.

48

PRODUCTION COMPUTER SYSTEM CONFIGURATION	1. Information Processing System (or Component)

Indicate the equipment components used in developing the above information processing system. If different systems with different configurations were used, or if similar systems with different configurations were implemented, consider each configuration as a separate system. For example, the same computer equipped with disc in one case and tape in another should be considered different configurations.

Configuration I				Configuration II				Configuration III			
2. Open Shop ___ 3. Closed Shop ___ 4. TS ___ 5. Number of Shifts _____ 6. # Operators/Shift 1__ 2__ 3__				Open Shop ___ Closed Shop ___ TS ___ Number of Shifts _____ # Operators/Shift 1__ 2__ 3__				Open Shop ___ Closed Shop ___ TS ___ Number of Shifts _____ # Operators/Shift 1__ 2__ 3__			
7. Eqpt	8. Mnfr	9. Model #	10. # of Units	Eqpt	Mnfr	Model #	# of Units	Eqpt	Mnfr	Model #	# of Units
Console				Console				Console			
CPU				CPU				CPU			
Add'l Core				Add'l Core				Add'l Core			
Data Channel				Data Channel				Data Channel			
File Control				File Control				File Control			
Disc Storage				Disc Storage				Disc Storage			
Drum Storage				Drum Storage				Drum Storage			
Tape Synch.				Tape Synch.				Tape Synch.			
Tape Drives				Tape Drives				Tape Drives			
Reader				Reader				Reader			
Printer				Printer				Printer			
_____				_____				_____			
_____				_____				_____			
_____				_____				_____			
_____				_____				_____			
_____				_____				_____			

FIGURE 8

SAMPLE COMPUTER DESCRIPTORS SUMMARY

3. Closed Shop--an installation where the programmers do not have "hands on" access to the computer; all runs are submitted to an operator.

4. TS (Time-Sharing)--specify if the program is being <u>developed on a time-sharing system</u> using interactive programming techniques (as opposed to developing a time-sharing system).

5. Number of Shifts--the number of shifts being used in program development. This figure may change as the development progresses; it is not necessary to complete another form if the system configuration stays the same; however, the change in the number of shifts should be indicated by means of a supplementary memorandum to be attached to this form indicating the change.

6. Operator/Shift--the number of computer operators per shift for each configuration.

7. Equipment--the components that make up the hardware system.

8. Manufacturer--the manufacturer of each hardware component in in the system.

9. Model Number--the identifying model number of each hardware component in the system.

10. Number of Units--the number of units of each component in the hardware system, e.g., eight high-density tapes, two typewriters, etc.

(If a diagram of the equipment layout is readily available, this should be attached to this form.)

d. <u>Personnel Data (Figure 9)</u>. The Personnel Data Form is used to record the pertinent background and experience of all personnel assigned to and charging to the development of computer program components in the system.

This form, filled out by staff personnel, should be handled as sensitive data, since it contains several personal items, such as test scores and salary. The data on this form are needed in the data bank, to analyze the effects of various personnel characteristics on programming costs. The absence of proficiency standards for programmer and analyst personnel has prompted the suggestion for various surrogates, which appear as items on this form.

In most organizations, many of the data items requested on this form would be found in a central personnel file. If such a central file was kept current, this form would not be necessary; only the names or identifying numbers of personnel would be needed, together with any of the data not found in the central file.

[1]Name or ID #	[2]Date of Birth	[3]Education	[4]Title and/or Job Duty	[5]Yrs in EDP/ Yrs w/org	[6]# Yrs Total Prog. Exp.	[7]Lang. Exp.	[8]Comp Exp.	[9]Appli- cation Exp.	[10]Periph. Exp.	[11]Formal EDP Training	[12]CDP Cert.	[13]EDP Test Scores	[14]Ranking	[15]Salary	[16]Remarks

FIGURE 9

SAMPLE PERSONNEL DATA FORM

This form should be periodically updated through the issuance of addenda to indicate any changes in the personnel assigned to the particular development effort, and to keep note of any changes in the status of those people originally assigned to the task.

The information solicited by this form includes:

1. Name--if employees have identification or code numbers, give those in place of names.

2. Date of birth.

3. Education--all undergraduate and graduate degrees, certificates, diplomas, and any additional courses; do not include formal EDP training, unless a full course of study was undertaken, e.g., B.S., Computer Sciences.

4. Title and/or Job Duties--principal job function performed by the employee, e.g., coding, flowcharting, design, analysis, etc.

5. Number of Years in EDP/Number of Years with organization--total length of employment (in years) in EDP industry including present employment, indicating number of years with present firm.

6. Number of Years' Total Programming and/or Analysis Experience--the portion of total employment in EDP (Question 5) which was spent in programming and/or analysis.

7. Language Experience--list all computer programming languages, procedure- and machine-oriented, with which employee is experienced; indicate length of experience in months, e.g., FORTRAN (6), COBOL (13), etc.

8. Computer Experience--list all computers with which employee is experienced and indicate length of experience in months with each machine, e.g., IBM 1401 (9), GE 225 (16), etc.

9. Application Experience--indicate the types of applications and length of experience (in months) with each, e.g., Math or Scientific Programming (8), Inventory Control (6), Compiler Development (13), etc.

10. Peripheral Experience--list the peripheral equipment (exclude printer) with which employee is experienced and length of experience in months with each; e.g., tapes (16), disc file (4), RAND tablet (7), CRT (1), etc.

11. Formal EDP Training--all formal EDP training (other than college degrees possibly recorded in 3 above) that the employee has received at his present or previous place of employment, e.g., manufacturer's school, company training program, etc.

12. CDP Certificate--indicate whether employee holds the Data Processing Management Association Certificate in Data Processing.

13. Test Scores--identify all tests and corresponding scores for all examinations the employee has taken to gain employment in the EDP field or to attain promotion from the job level to another.

14. Ranking--if employees are ranked (or evaluated) by supervisor either on a staff basis or a company basis, indicate ranking.

15. Salary--indicate starting and present salary for the employee.

16. Remarks--any pertinent remarks that may be of value in judging the employee's proficiency; e.g., discrepancy between job performance and test scores, etc.

e. Computer Program Component Characteristics Summary (Figure 10). The data requested in this form (shown in Figure 10) are intended to describe the Computer Program Component as a product in terms of numerical measures that can be applied to a wide variety of computer programs. These data can be obtained from a computer listing of the completed program, interviews with the lead programmer, and abstracts of documentation. One form should be completed for each computer program component in the information processing system.

The Computer Program Component Characteristics Form consists of the following items:

1. Computer Program Component--name or identification number of the computer program to be described on this form.

2. Information Processing System--name of the information processing system of which the computer program (Item 1) is a component.

3. Source Language(s)--the source language(s) used in developing the component.

4. Number of Subroutines--number of subroutines, including library routines, the computer program component contains.

5. Number of Unconditional Branch Object Instructions--number of unconditional branch object (machine or assembly language) instructions the completed computer program component contains.

COMPUTER PROGRAM COMPONENT CHARACTERISTICS SUMMARY	DATE:

1. Computer Program Component:	2. Information System Name:	3. Source Languages:

4. Number of Subroutines:	5. Number of Unconditional Branch Object Instructions:	6. Number of Conditional Branch Object Instructions:

7. List all programs that supply input or receive output from this program:

8. List all programs that have an operational interface with this program:

9. Internal Documents

Document Name	Number of Pages

10. External Documents

Document Name	Number of Pages

11-16 Type of instructions	Number of Source Instructions	Number of Object Instructions
11-12 Number of delivered instructions	11.	12.
13-14 Number of delivered instructions written expressly for this program	13.	14.
15-16 Number of instructions written on generated but not delivered	15.	16.

17. Describe in detail the functions that this program performs, and its place in, and relation to, the information processing system of which it is a part.

FIGURE 10

SAMPLE COMPUTER PROGRAM COMPONENT CHARACTERISTICS SUMMARY

6. Number of Conditional Branch Object Instructions--Number of conditional branch object instructions the completed computer program component contains.

7. List of computer program components that supply input to or receive output from this program component. (Include all computer program components, i.e., in the information processing system of which this component is a part, as well as any other components from other information processing systems.)

8. List of all interfaces other than data input or output, that this program component has with other program components, both in this or any other information processing system.

9. List of internal document types with corresponding number of pages for each--distinct internal documents that were developed in the project during the development of the program. Note: An internal document is one that is used by the project staff in program development and which is not for outside release. A page is defined as one single-spaced side of an $8\frac{1}{2}$" by 11" printed sheet.

10. List of number of external document types--how many distinct documents does the project have to produce that will be used outside of the programming organization, e.g., training manuals and system and program specifications.

11. Number of delivered source instructions--the number of instructions in the program at time of turnover to the customer. Note: If more than one source language was used, e.g., both POL and MOL, indicate the number of instructions for each language.

12. Number of delivered object instructions--the total number of object instructions in the computer program component at time of turnover to the customer.

13. Number of source instructions written for this computer program component--of the total number of source instructions indicated in Item 11, how many were specifically written for this program component? Note: The difference between Item 11 and Item 13 will be the number of source instructions borrowed from existing programs and/or library routines; these are included in Item 11.

14. Number of delivered object instructions written for this computer program component--of the delivered instructions indicated in Item 12, how many were written specifically for this program component? Note: The difference between Item 12 and Item 14 will be the number of object instructions borrowed from existing programs and/or library routines; these are included in Item 12.

15. Number of source instructions written but not delivered--the total number of source instructions written during program production that were not included as part of the completed program component at time of turnover to the customer. Note: This would include instructions in test programs, debugging aids, and other support and utility programs that were needed during program production, but were not part of the delivered program.

16. Number of object instructions written but not delivered--the total number of object instructions generated during program production that were not a part of the completed program component at time of turnover to the customer. Note: This would include instructions in test programs, debugging aids, and other support and utility programs that were needed during program production, but were not part of the delivered program.

17. Program function--describe in detail the functions that this program performs in the information processing system of which it is a part. Indicate the interfaces, required input and outputs and timing specifications. Also, specify the approximate percentage of the total computer program devoted to each function that the program is required to perform. (This information could be taken from an abstract for the operational design specification for this component.)

f. Development Environment. This form (Figure 11) requests data for items that characterize the organizational and management environment in which the information processing system was developed.

If the information processing system contains several computer program components, and the development environment is different for some of those components, a form like this should be completed for each.

The following items characterize the development environment:

1. Computer program component--name and identification number of the computer program.

2. Information processing system--name of the information processing system of which Item 1 is a part.

3. Was this program the first one to be produced on the developmental computer?

4. List of all organizations (or agencies) that have to concur on computer program component and/or information processing system design.

57

DEVELOPMENT ENVIRONMENT SUMMARY	DATE:

1. Computer Program Component: 2. Information Processing System:	3. First Program on Developmental computer: Yes _____ No _____
4. List of organizations using the computer program component: _____ _____ The information processing system _____ _____ _____	5. List of Concurring Organizations: _____ _____ _____ _____ 6. Distance in miles between operational and developmental computers:
7. Developmental computer controlled by developmental agency? Yes _____ No _____	9. Estimated EDP development experience of user: Extensive _____ Limited _____ None _____
8. Schedule constraints: Dependent upon Yes ___ No ___ Depended on Yes ___ No ___ Tight _____ Loose _____ Average _____	
10. List all geographical locations used during program development: _____ _____ _____ _____ _____ _____	11. List the number of trips during program development and the round-trip distance of each trip: _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____
12. Turnaround time: Minimum _____ Maximum _____ Typical _____	13. Extent of review and approval

FIGURE 11

SAMPLE DEVELOPMENT ENVIRONMENT SUMMARY

5. List of the organizations with different missions or functions that require the data processing support provided by this computer program component, and the information processing system of which it is a part.

6. If the developmental and operational computers are not in the same geographical location, indicate how far apart they were, in miles.

7. Indicate whether the agency responsible for program development had control over the developmental computer.

8. Indicate whether the schedule for this system development and installation was dependent upon, and depended on, schedules for other developments (e.g., information systems, computers, communications). Indicate whether the schedule was regarded as tight, loose, or average.

9. Classify the customer's (requesting agency's) experience and knowledge concerning the development of automatic data processing systems as: extensive, limited, or none.

10. List the different geographical locations where the program was developed.

11. State how many man trips were necessary during program development, and the average round-trip distance per trip.

12. Indicate the range and typical turnaround time experienced by the programmers assigned to this project. Note: Turnaround time is the time span between submitting and receiving computer runs, such as a compile, debug, test, during the development of the computer program component.

13. Identify each review and approval performed by personnel outside the development effort and "turnaround" time (period from transmittal of request for review until approval or action-triggering feedback is received) for each of these reviews.

g. The Development History. The end product of the cost-collection system presented in this report is the Development History. This history contains the completed forms that constitute the complete cost and progress documentation of the computer program development process for the period during which the work was performed.

Eight types of collection forms constitute the Development History:

- The Summary Weekly Activity Report (Figure 3)

- The Systems Requirements Summary (Figure 7)

- The Computer Descriptors Summary (Figure 8)

- The Personnel Data Form (Figure 9)

- The Computer Program Component Characteristics Summary (Figure 10)

- The Development Environment Summary (Figure 11)

- Information Processing System Progress and Costs (Figure 4)

- The IPS Progress Graphs (Figure 5)

In addition to these major forms, the history file also contains all changes and additions to the above forms, so that a complete description of all phases of the development effort is contained in this history. This history can be assembled when the products are turned over for use. Updates that reflect work done in the maintenance step may be added at a later time. The Development History is designed to supply uniform data for many different development efforts as inputs for analysis conducted to answer the questions that pertain to costs and schedules such as those listed earlier in this section.

The data in the completed forms should be regarded as raw data that would need at least coding to be used in analysis and further research to be completed, in some cases. Thus, for each completed effort the qualitative and quantitative data are entered in a data bank that is available to the researcher, but the organization of these data for actual analysis is left primarily to his discretion. If it is found that a large amount of data are to be stored and retrieved for analytical work, careful study should be devoted to the possibility of automating the data bank operations. Further, in considering the specific application of this system to any particular organization, some thought and design effort should be devoted to coding the data at the source and automating parts of the recording and collection procedure.

SECTION V

DATA-COLLECTION PROCEDURES AND RESPONSIBILITIES

The previous sections, which described a planning structure for computer program development, along with a numbering system for relating parts of the work with recorded expenditures, defined: (1) the data to be collected, (2) the forms for recording these data and for subsequently summarizing these recorded data in data management reports, and (3) a way to assemble a set of these data into a complete development history as an entry into a data bank. This section supplies other parts of the system description, namely: (1) identification of duties for the individuals who would assume responsibility in operating the collection system, (2) the timing for completion of the data forms, and (3) their flow during a typical computer program development effort. Since this cost-collection system is advanced as a tool for use in a wide variety of organizations, each of which may pursue a spectrum of computer programming efforts, the responsibilities for operating the system, the duties, are grouped into position descriptions that may or may not correspond to an actual job being performed by an individual in a computer program development effort. These positions are typical of the staff management and technical jobs in large programming organizations. In a similar way, the timing and flow of the data have been generalized, but are intended to be easily matched with an actual computer program development effort.

1. Position Descriptions. The duties involved in the operation of this system for collecting data on costs and cost factors in computer program development have been divided among six positions. Before these positions are described, the rationale behind this particular division of labor is developed.

To minimize the burden upon personnel in the line organization that would be performing the technical work in computer programming, most of the duties in this cost-collection system have been assigned to a staff position. The location of this staff position within an organization is largely irrelevant; the duties could even be done effectively by an outside organization. The only real "authority" required by staff personnel involved in data collection is the right of access. In fact, within an organization the separation of the cost-collection effort from performance of the project (so that the staff personnel involved would not report to any line manager directly responsible for a computer programming project) could reduce any possible conflict of interest that might influence the data reliability (20).

The assignment of most duties in the collection of costs to a staff position does not eliminate the cost of collecting; however, this organizational arrangement has several advantages that help reduce these costs:

61

. It would involve lower costs to train personnel in the use of the system, since fewer people need to know the detailed procedures.

. Experience gained on different projects by the cost-collection staff would help them to gain proficiency; their specialization would tend to make them more productive.

. The time of scarce technical personnel would be conserved. Personnel selected for this cost-collection work should have a personal tolerance and aptitude for the administrative detail required by the cost-reporting system--traits that are not common among technical specialists.

The six positions involved in the operation of the cost-collection and -reporting system are the Staff Assistant; Computer Programming Lead Programmer or Analyst; Programmer or Systems Analyst; Project Manager (or the Officer who monitors for computer programming in a System Project Office (SPO) Officer); General Corporate Management; and finally, the Data Bank Librarian. Table I shows the duties for each of these positions. These positions are described in more detail below.

a. <u>The Staff Assistant</u>. The Staff Assistant is the coordinator and communication point in this cost-collection system. Successful operation of the system depends heavily upon him, since he alone has direct contact with every collection form in the system during the collection phase.

The Staff Assistant must have a thorough knowledge of the total cost-reporting system, its purposes, its operation, and its function within the financial structure of the performing organization. He should be well informed both technically and administratively. He must know enough about the technical content of the development for which the costs are being collected, so that he may easily and quickly pinpoint changes in the system requirements, project personnel, and program design that are required in the collection forms. He should know enough about basic accounting principles and the financial auditing procedures used in the organization to verify cost data.

b. <u>Lead Programmer (Analyst)</u>. The Lead Programmer, or Analyst, the first level of supervision to come in contact with a form in this collection system, is the technical head of a group of programmers and analysts. He assigns work, monitors progress, receives and dispatches the necessary changes in the programs, and is, in general, a technical troubleshooter for his group. In practice, this position in the cost-collection system could be filled by more than one person, because responsibility may shift within an organization as the project proceeds through various steps. Also, in developing a large system, there may be many Lead Programmers and Analysts.

TABLE I

DUTIES AND POSITIONS FOR THE COST-COLLECTION SYSTEM

System Position	System Duties

Staff Assistant	Collect data required, and prepare:
	    System Requirements Description Summary
	    Computer Descriptors Summary
	    Personnel Descriptors Summary
	
	Prepare updates to these as values for data items change
	
	Collect Weekly Activity Reports (WAR) and combine in Summary Weekly Activity Report (SWAR)
	
	Prepare periodic Information Processing System (IPS) Status Reports:
	    IPS Progress and Costs
	    SWAR
	    IPS Progress Graphs
	
	Prepare IPS Management Summary
	
	After installation step, prepare:
	    Computer Program Component Characteristics
	      Summary
	    Development Environment
	
	Assemble Development History and deliver to data bank

Lead Programmer (Analyst)	Review WARs for accuracy and completeness Review SWAR and add total instructions completed Approve management reports, review all other documents for accuracy
Programmer or Systems Analyst	Complete WAR

Project Manager and/or System Project Office	Receive IPS Status Report:
	    IPS Progress and Costs
	    SWAR
	    IPS Progress Graphs

Corporate Management	Receive IPS Management Summary
Data Bank Librarian	Receive and file Development History

The Lead Programmer must have a good working knowledge of the collection system since his task is to verify the accuracy of the programmers' and analysts' records of their time allocation (Weekly Activity Reports), the Staff Assistant's records of the system and program changes, and the other source data in the completed forms that will constitute the complete Development History.

   c. Programmers and Analysts. These positions in the cost-collection system are filled by members of the force assigned to work on the development of the information processing system. Their activities result in the expenditure of manpower that is recorded as inputs to the system in the Weekly Activity Reports. These men, the major data sources in the system, supply the most important information being collected. In the Weekly Activity Report, each Programmer or Analyst records how he allocates his time to each step (and task, possibly) in the development of a computer program. These technical personnel only need a minimal knowledge of the cost-collection system, but they must know how to identify the specific task they are working on with a specific step or work order number.

   d. Project Manager.[2] The Project Manager represents the first level of management that has authority to make changes in the development of the information processing system for other than purely technical reasons. The scope of his responsibility depends upon the nature of the particular information processing system under development. For a large system with many components, there may be intermediate management levels that would receive reports on the components.

This level of management receives three summary reports indicating the progress of the project with respect to schedules and resources. The comparison between actuals and estimates in these forms permit him to assess easily the progress of the project. If the Project Manager is not satisfied with the activities or progress in the work, he is in a position to make changes with regard to costs and staffing, and to put the project back on its desired course.

_____

[2] In the Air Force, the System Program Office (SPO) is the lowest level of USAF management charged with the overall responsibility for the performance of work on a specific project; it must assure that performance meets requirements. The SPO may or may not have influence in the determination of the requirements and the design of the project. This cost-reporting system would deliver identical progress reports to the SPO (in particular the officer responsible for monitoring the computer programming) and the Project Manager within an implementing organization.

e. Corporate Management. The term Corporate Management is used in the cost-reporting system to represent a corporation officer who is responsible for the total financial expenditure of the company. This position could also correspond to the financial officer or comptroller in a government agency.

This manager receives a summary of the Programming Project Management Summary indicating expenditures to date and the projected expenditures for the job to completion.

His position and authority offer him the opportunity to control corporate resources, and, if the project runs into difficulty, he may make the necessary changes, such as appropriating additional funds or ordering cutbacks on certain resources.
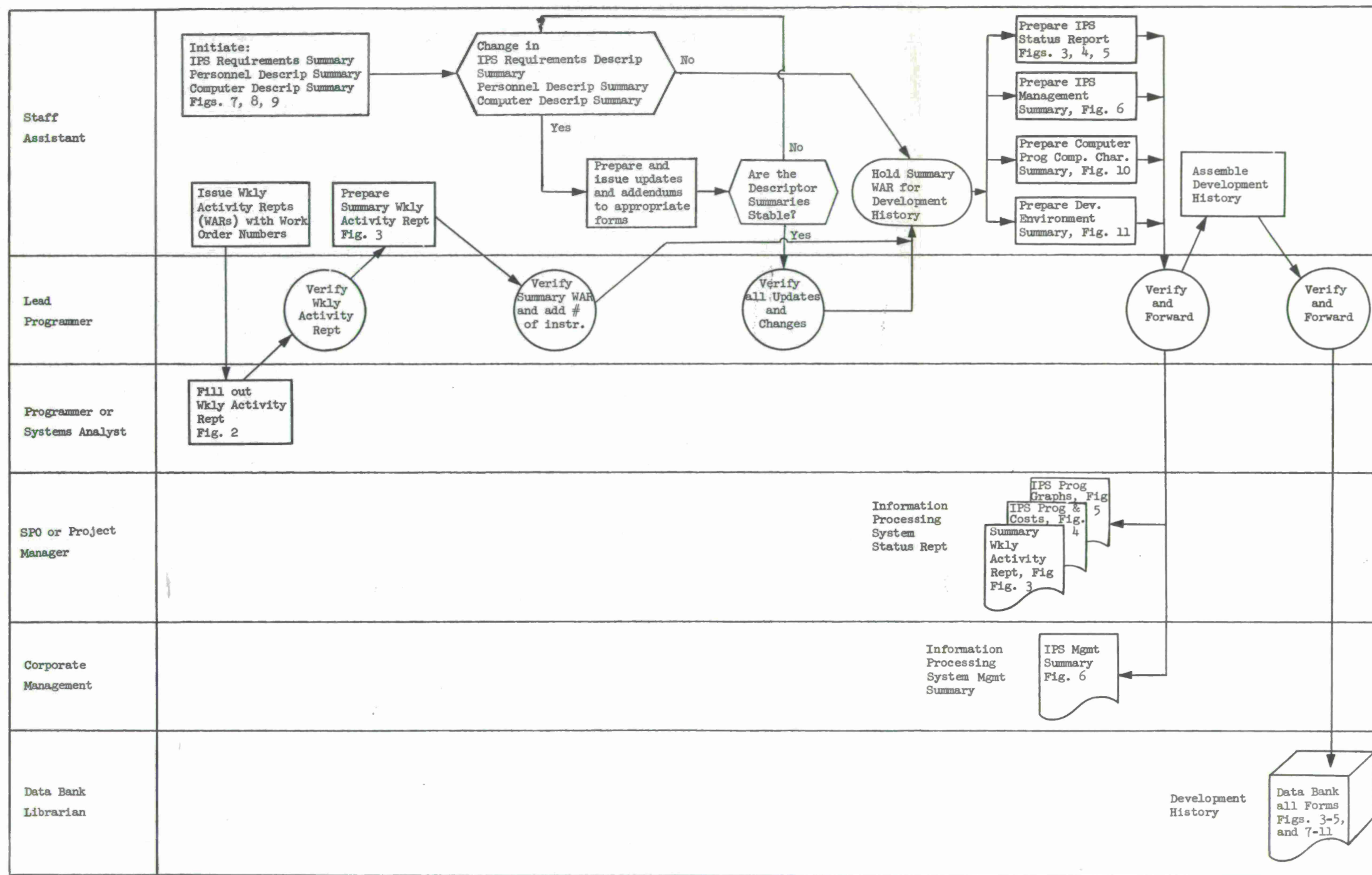
f. Data Bank Librarian. The Data Bank Librarian performs storage and retrieval functions for the repository of histories for completed projects. The Development History is made up of the final version of all forms in the collection system, with any other documentation relevant to the history of a computer programming project.

The Data Bank could be administered by a full-time archivist or an employee assigned to the job part time, depending on the size of the organization, and the expected activity in the storage and retrieval of cost data.

2. The Work Flow. The cost-collection and summary reporting forms described earlier are the foundation of this cost-reporting system. To operate the system successfully, the Staff Assistant must carefully coordinate a sequence of work tasks. This sequence of activities is the work and data flow of the cost-reporting system. Figure 12 illustrates one possible flow for the forms and also shows the corresponding duties that would be performed in operating this cost-collection system. This is only one way to set up the system; there are many possible alternatives depending on the needs of the organization that adopts and uses the system.

The operation of the cost-collection system begins after the first step in the computer programming process, Feasibility Analysis, when the decision is made to proceed with an identifiable development effort. After the "go ahead" decision is made, staffing begins for the organization that will be responsible for developing the information system. Before the personnel are assigned to carry out the work, the work order or charge numbers to be used in the system are assigned, so that each person can begin charging the budget allocated to the development of the system.

The Staff Assistant prepares the blank Weekly Activity Reports and issues these to each person assigned to the effort. He inserts appropriate work order numbers. He may also insert the corresponding step or task name, thus eliminating the need for the analysts and programmers involved to remember what steps each work order number represents.

Staff Assistant

Initiate:
IPS Requirements Summary
Personnel Descrip Summary
Computer Descrip Summary
Figs. 7, 8, 9

Change in
IPS Requirements Descrip
Summary
Personnel Descrip Summary
Computer Descrip Summary — No

Yes

Issue Wkly Activity Repts (WARs) with Work Order Numbers

Prepare Summary Wkly Activity Rept Fig. 3

Prepare and issue updates and addendums to appropriate forms

Are the Descriptor Summaries Stable? — No

Yes

Hold Summary WAR for Development History

Prepare IPS Status Report Figs. 3, 4, 5

Prepare IPS Management Summary, Fig. 6

Prepare Computer Prog Comp. Char. Summary, Fig. 10

Prepare Dev. Environment Summary, Fig. 11

Assemble Development History

Lead Programmer

Verify Wkly Activity Rept

Verify Summary WAR and add # of instr.

Verify all Updates and Changes

Verify and Forward

Verify and Forward

Programmer or Systems Analyst

Fill out Wkly Activity Rept Fig. 2

SPO or Project Manager

Information Processing System Status Rept

IPS Prog Graphs, Fig. 5
IPS Prog & Costs, Fig. 4
Summary Wkly Activity Rept, Fig. 3

Corporate Management

Information Processing System Mgmt Summary

IPS Mgmt Summary Fig. 6

Data Bank Librarian

Development History

Data Bank all Forms Figs. 3-5, and 7-11

Note: Figure numbers refer to figures in Section IV.

FIGURE 12

COST-COLLECTION AND -REPORTING SYSTEM WORK FLOW

The Weekly Activity Reports are issued, completed, and collected at weekly intervals. Then they are checked for accuracy by the Lead Programmer or Analyst and passed back to the Staff Assistant.

The Staff Assistant summarizes, transfers, and verifies the transfer of the information from this form to the Summary Weekly Activity Report, and discards the individual activity reports unless they are needed for other purposes, such as accounting. The Summary Weekly Activity Report then becomes a comprehensive account of the work activities for the previous week. This form is held for incorporation into the Information Processing System Status Report and the Development History. As soon as personnel charge to a work order number, the Staff Assistant records data for them on the Personnel Description Summary. This form will contain the personnel history of the programmers, analysts, and support personnel assigned to work on the system and its components. The Staff Assistant will keep this form until the end of the development cycle, so that he can make the appropriate additions as personnel changes occur.

The next phase in the work flow is the compilation of information for the System Requirements Summary, and the Development Computer Description. To supply information for these two forms the Staff Assistant uses outputs from the second and third steps in the Computer Programming Process: Information Processing Analysis and Information Processing Design. It is assumed that these outputs are documented--at least in draft form. At this early stage these two forms on requirements and computers, as well as the Personnel Description Summary, are likely to be incomplete, and additions will have to be made later. Further, the System Requirements are likely to change before this information becomes stable. Changes or additions should be in the form of updates or addenda to the original forms documented by the Staff Assistant, so that a complete history of all changes can be compiled for future study. (For a more accurate history, changes to existing data could be differentiated from addenda to complete forms with missing data.) The changes to the forms should be verified by the Lead Analyst and returned to the Staff Assistant.

Based upon a decision made by the Project Manager and Corporate Management on the frequency and detail with which they want to be kept abreast of the project's progress, the Staff Assistant compiles the two periodic management summaries, namely, the Information Processing System Status Report and the Information Processing System Management Summary. To prepare these forms, he uses the information taken from the Summary Weekly Activity Reports.

The Information Processing System Status Report is based upon information from three forms--the Summary Weekly Activity Report, the Information Processing System Progress and Costs Summary, and the Information Processing System Progress Graphs. These completed forms supply the Project Manager with all the data needed to monitor the work progress with respect to schedules and costs.

69

The Information Processing System Management Summary is prepared for the benefit of Corporate Management. These data are intended to help the corporate officer who is responsible for the expenditure and allocation of funds and include an accumulated total of actual and estimated expenditures for the project.

The Staff Assistant compiles and forwards these management summaries to the Lead Programmer(s) or Analyst(s) for verification. If the information is verified, the Staff Assistant sends the forms to the appropriate manager; if not, the data are appropriately modified by the Staff Assistant. This procedure is repeated at the intervals specified by the recipients of the management reports.

At the completion of the computer program, two remaining forms are filled out, the Component Characteristics Summary, and the Development Environment Summary, to describe the characteristics of the resulting product, the computer program, and the environment in which it was developed. This information is compiled for inclusion in the Development History.

Table II summarizes the responsibilities associated with each form by indicating the time of completion, the responsible position, and the purpose of the data. At first glance, this reporting system may seem cumbersome, due to the number of forms required; however, with the exception of the Staff Assistant, the workload burden is minimal, as indicated by Figure 13, for all persons directly involved in the production of the final product, the computer program. In addition, if the organization using this system does not wish to compile extensive data for research, the forms to record data on requirements, personnel, computer environment, and computer program components characteristics may be trimmed or even eliminated from the system, without destroying its management control and reporting capabilities.

TABLE II

SUMMARY OF DATA COLLECTION FORMS

SCHEDULE, POSITION, AND PURPOSE FOR COMPLETION OF COLLECTION FORMS			
Form Name	When Prepared	Prepared by	Purposes
Weekly Activity Report	Weekly	Programmer, Systems Analyst, and every direct worker	Source cost data acquisition
Summary Weekly Report	Weekly	Staff Assistant	Reduce volume of paper produced by Weekly Activity Reports; summarize data for management review
System Requirements Summary	Before Start of Systems Analysis and After Changes	Staff Assistant	Describe function and purpose of computer program. Define and list those characteristics imposed upon the computer program by its performance specifications
Computer Description Summary	Before Start of Systems Analysis and After Changes	Staff Assistant	Define the salient characteristics of the computer facility used to develop the computer program
Personnel Description Summary	Before Start of Systems Analysis and After Changes	Staff Assistant	Define characteristics of the direct personnel working on project
Computer Program Component Characteristics	After Program Installation	Staff Assistant	Describe and summarize salient features of the computer program listing
Program Environment Summary	After Program Installation	Staff Assistant	Gather and record data on factors that characterize the surrounding in which the work is done

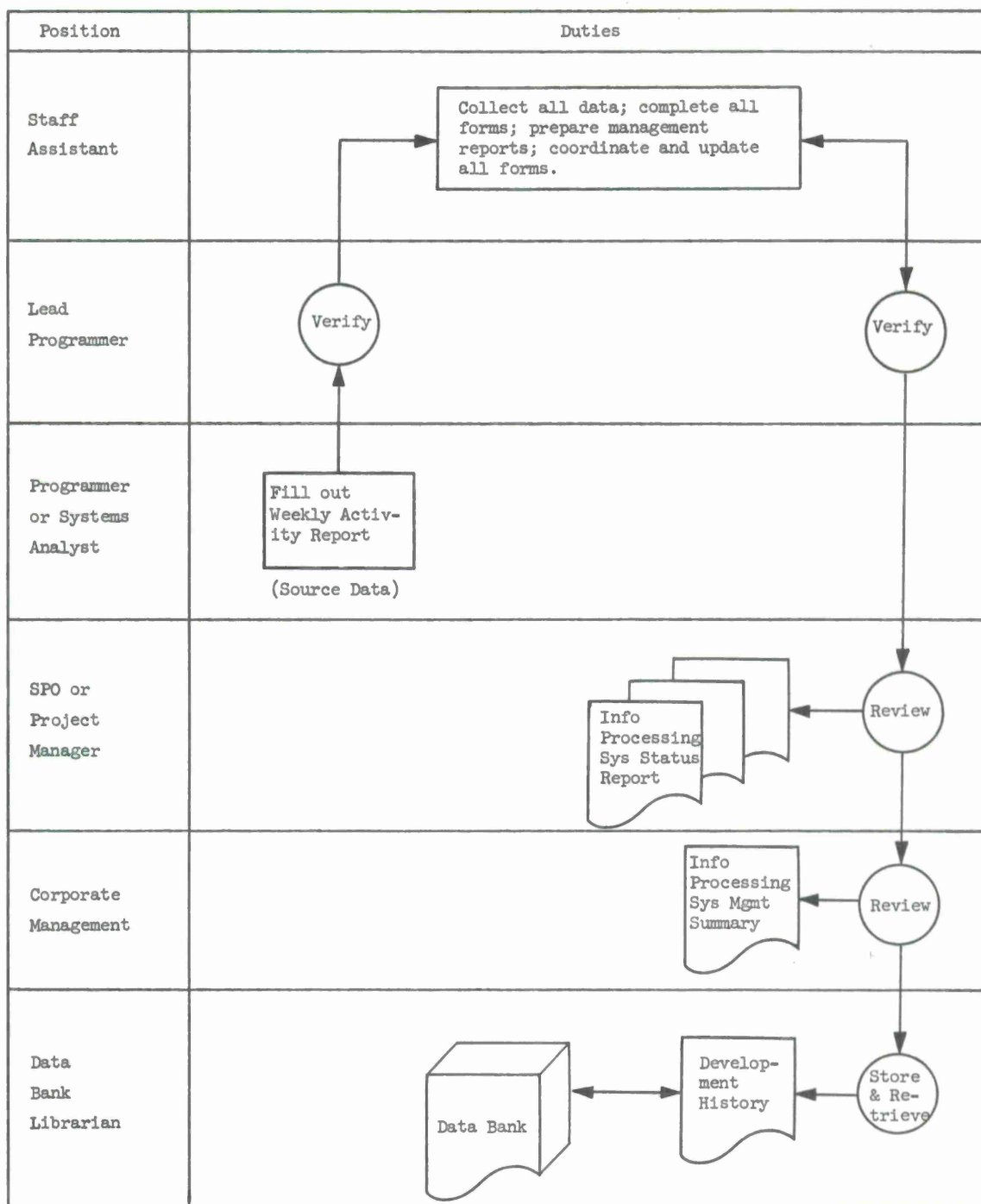| Position	Duties
Staff Assistant	Collect all data; complete all forms; prepare management reports; coordinate and update all forms.
Lead Programmer	Verify Verify
Programmer or Systems Analyst	Fill out Weekly Activity Report (Source Data)
SPO or Project Manager	Info Processing Sys Status Report Review
Corporate Management	Info Processing Sys Mgmt Summary Review
Data Bank Librarian	Data Bank Development History Store & Retrieve

FIGURE 13

INPUT/OUTPUT TRAIL IN THE WORK FLOW OF THE COST-REPORTING SYSTEM

REFERENCES

1. G. F. Weinwurm, "Data Elements for a Cost Reporting System for Computer Program Development." TM-2934/000/02, System Development Corporation. 6 July 1966.

2. L. Farr, V. LaBolle, and N. E. Willmorth, "Planning Guide for Computer Program Development." TM-2314/000/00, System Development Corporation. 10 May 1965.

3. E. A. Nelson, "Management Handbook for the Estimation of Computer Programming Costs." TM-3225/000/01, System Development Corporation, 20 March 1967.

4. Data Automation--Supporting Data Systems. Air Force Regulation 300-3, Department of the Air Force, Washington, D. C. 1 September 1964.

5. D. H. Brandon, Management Standards for Data Processing. D. Van Nostrand Company, Inc., Princeton, New Jersey. 1963.

6. Report to the President on the Management of Automatic Data Processing in the Federal Government. Bureau of the Budget, Washington, D. C. 4 March 1965

7. K. Heinze, N. Claussen, and V. LaBolle, "Management of Computer Programming for Command and Control Systems." TM-903/000/02, System Development Corporation. 8 May 1963.

8. R. E. Bleier, "A Description of the Computer Program Implementation Process: A Process Flow." TM-1021/003/00, System Development Corporation. 9 May 1963.

9. L. Farr and H. J. Zagorski, "Factors that Affect the Cost of Computer Programming: A Quantitative Analysis." TM-1447/001/00, System Development Corporation. 31 August 1964.

10. E. A. Nelson, "Research Into the Management of Computer Programming: Some Characteristics of Programming Cost Data from Government and Industry." TM-2704/000/00, System Development Corporation. 15 November 1965.

11. G. F. Weinwurm and H. J. Zagorski, "Research Into the Management of Computer Programming: A Transitional Analysis of Cost Estimation Techniques." TM-2712/000/00, System Development Corporation. 12 November 1965.

12. V. LaBolle, "Development of Equations for Estimating the Costs of Computer Program Production." TM-2918/000/00, System Development Corporation. 5 April 1966.

13. T. Fleishman, "Current Results from the Analysis of Cost Data for Computer Programming." TM-3026/000/01, System Development Corporation. 27 June 1966.

14. Configuration Management Exhibit for Computer Programs, ESD Exhibit EST-1, and Systems Management Procedures for Computer-Based Command and Control Systems, ESD Exhibit EST-2. Technical Requirements and Standards Office, Electronic Systems Division, Air Force Systems Command, Bedford, Massachusetts. Also see: L. V. Searle, G. Neil, and S. G. Benson, "Supplementary ESD Narrative Material (for) AFSCM 375-4," TM-2661/000/01, System Development Corporation, 27 June 1966; L. V. Searle, G. Neil, and R. L. Henderson, "Systems Management Documentation for Computer Programs." TM-3014/000/00, System Development Corporation, 17 June 1966.

15. Armed Forces Management. October 1966, page 100.

16. Reducing Contractor Reporting Requirements. Logistics Management Institute, Washington, D. C. November 1962, page 17.

17. Planning-Programming Budgeting. Bulletin 66-3, Bureau of the Budget, Washington, D. C. 12 October 1965.

18. Cost Estimating Procedures. AFSCL 173-1, Air Force Systems Command, Andrews Air Force Base. 1 October 1965.

19. Cost Information System. AFSCL 173-2, Air Force Systems Command, Andrews Air Force Base. 1 October 1965.

20. H. Koontz and C. O'Donnell, Principles of Management. McGraw-Hill Book Company, New York. Third Edition, 1964.

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*	2a. REPORT SECURITY CLASSIFICATION
System Development Corporation 2500 Colorado Avenue Santa Monica, California 90406	Unclassified
	2b. GROUP N/A

3. REPORT TITLE

A SYSTEM FOR COLLECTING AND REPORTING COSTS IN COMPUTER
PROGRAM DEVELOPMENT

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

None

5. AUTHOR(S) *(First name, middle initial, last name)*

Nelson, Edward A.
Fleishman, Thomas

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
11 April 1967	78	20

8a. CONTRACT OR GRANT NO. AF 19(628)-67-C-0132	9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-67-452
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* TM-3411
d.	

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Computer Directorate, Electronic Systems Division, Air Force Systems Command, USAF, L. G. Hanscom Field, Bedford, Mass. 01730

13. ABSTRACT   This report describes a system for the collecting and reporting on data on the resources expended in the production of computer programs. The system is intended to: (1) provide information to facilitate management control during the progress of a computer programming effort; (2) build a data bank from which better cost-estimating relationships and planning tools can be developed; (3) accomplish the above with a minimum of interference with operating personnel. The report was designed to provide sample materials necessary for the implementation of cost reporting in any organization in which computer programming is performed; it includes a description of the steps that constitute the computer programming process, the kinds of personnel who would be involved in the cost-collection and -reporting system, a recommended work flow and suggested forms for use in data collection and reporting, a work breakdown structure for associating costs with activities, and a brief discussion of the relationship of this system with several existing Department of Defense management procedures.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computer Programming -- Costs						
Management Systems						
Planning						
Cost-Collecting						
Cost-Reporting						

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (corporate author) issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the over-all security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accord-ance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD direc-tive 5200.10 and Armed Forces Industrial Security Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classifica-tion, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of the author(s) in normal order, e.g., full first name, middle initial, last name. If military, show grade and branch of service. The name of the principal author is a minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the re-port, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the num-ber of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, and 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, task area number, systems numbers, work unit number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers (either by the originator or by the sponsor), also enter this number(s).

10. DISTRIBUTION STATEMENT: Enter the one distribution statement pertaining to the report.

Contractor-Imposed Distribution Statement

The Armed Services Procurement Regulations (ASPR), para 9-203 stipulates that each piece of data to which limited rights are to be asserted must be marked with the following legend:

"Furnished under United States Government Contract No._____ . Shall not be either released outside the Government, or used, duplicated, or disclosed in whole or in part for manufacture or procurement, without the written permission of _____ , except for: (i) emergency repair or overhaul work by or for the Government, where the item or process concerned is not otherwise reasonably available to enable timely performance of the work; or (ii) release to a foreign government, as the interests of the United States may require; provided that in either case the release, use, duplication or disclosure hereof shall be subject to the foregoing limitations. This legend shall be marked on any reproduction hereof in whole or in part."

If the above statement is to be used on this form, enter the following abbreviated statement:

"Furnished under U. S. Government Contract No._____ . Shall not be either released outside the Government, or used, duplicated, or disclosed in whole or in part for manufacture or procurement, without the written permission of_____ , per ASPR 9-203."

DoD Imposed Distribution Statements (reference DoD Directive 5200.20) "Distribution Statements (Other than Security) on Technical Documents," March 29, 1965.

STATEMENT NO. 1 - Distribution of this document is unlimited.

STATEMENT NO. 2 (UNCLASSIFIED document) - This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of (fill in controlling DoD office).

(CLASSIFIED document) - In addition to security require-ments which must be met, this document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval (fill in controlling DoD Office).

STATEMENT NO. 3 (UNCLASSIFIED document) - Each trans-mittal of this document outside the agencies of the U. S. Government must have prior approval of (fill in controlling DoD Office).

(CLASSIFIED document) - In addition to security require-ments which apply to this document and must be met, each transmittal outside the agencies of the U. S. Government must have prior approval of (fill in controlling DoD Office).

STATEMENT NO. 4 (UNCLASSIFIED document) - Each trans-mittal of this document outside the Department of Defense must have prior approval of (fill in controlling DoD Office).

(CLASSIFIED document) - In addition to security require-ments which apply to this document and must be met, each trans-mittal outside the Department of Defense must have prior ap-proval of (fill in controlling DoD Office).

STATEMENT NO. 5 (UNCLASSIFIED document) - This document may be further distributed by any holder only with specific prior approval of (fill in controlling DoD Office).

(CLASSIFIED document) - In addition to security require-ments which apply to this document and must be met, it may be further distributed by the holder ONLY with specific prior approval of (fill in controlling DoD Office).

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (paying for) the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical re-port. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified re-ports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as (TS), (S), (C), or (U).

There is no limitation on the length of the abstract. How-ever, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Iden-tifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of techni-cal context. The assignment of links, roles, and weights is optional.